

Rec'd PCT/PTO 18 APR 2005

PCT/EP 10033179P8 #2

BUNDESREPUBLIK DEUTSCHLAND

28 JAN 2004

**PRIORITY
DOCUMENT**

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)



REC'D 19 FEB 2004

WIPO PCT

**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung**

Aktenzeichen:

102 50 148.3

Anmeldetag:

28. Oktober 2002

Anmelder/Inhaber:

Océ Printing Systems GmbH, 85586 Poing/DE

Bezeichnung:

Verfahren und System zum Erzeugen einer
grafischen Benutzeroberfläche zum Bedienen,
zum Konfigurieren und/oder zur Diagnose eines
elektrofotografischen Druckers oder Kopierers

IPC:

G 03 G 15/00

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 22. Januar 2004
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Klostermeyer

Beschreibung

Verfahren und System zum Erzeugen einer grafischen Benut-
zeroberfläche zum Bedienen, zum Konfigurieren und/oder zur
5 Diagnose eines elektrofotografischen Druckers oder Kopie-
rers

Die Erfindung betrifft ein Verfahren und ein System zum
Erzeugen einer grafischen Benutzeroberfläche für ein elek-
10 trofotografisches Druck- oder Kopiersystem. In einem Spei-
cherbereich einer ersten Datenverarbeitungseinheit des
Druck- oder Kopiersystems sind Daten zum Erzeugen einer
grafischen Benutzeroberfläche gespeichert.

15 Bei bekannten Druckern oder Kopierern ist eine serielle
Schnittstelle zum Anschluss eines Service- und War-
tungscomputers vorgesehen, mit dem Diagnosearbeiten und
Einstellarbeiten durchgeführt werden können. Einzelne
Baugruppen des Druckers oder Kopierers haben eine eigene
20 Datenschnittstelle, an die über einen Schnittstellenwand-
ler der Service- und Wartungscomputer für Diagnose-, Kon-
figurations- und Wartungsarbeiten anschließbar ist. Die
Diagnose und Einstellung dieser Baugruppen kann bei be-
kannten Druckern nur über die jeweilige Datenschnittstelle
25 der Baugruppe erfolgen. Durch die relativ geringen Daten-
übertragungsraten dieser Datenschnittstellen mussten grö-
ßer Datenmengen mit Hilfe von Wechseldatenträgern, wie
z.B. Disketten, aus dem Drucker oder Kopierer ausgelesen
werden. Zum Auswerten dieser Fehlerdaten, zum Diagnosti-
30 zieren und Warten der Baugruppen mit eigener Datenschnitt-
stelle und zur Diagnose des Druckers oder Kopierers über
die serielle Schnittstelle sind unterschiedliche Programme
erforderlich, die jeweils durch den Service- und War-
tungscomputer einzeln aufgerufen und abgearbeitet werden.
35 Die erforderlichen Programme sind auf dem Service- und
Wartungscomputer installiert, wobei für einzelne Ausliefe-
rungsstände einzelner Baugruppen und des Druckers oder

Kopierers unterschiedliche Programmversionen erforderlich sind, die jeweils als separates Programm auf einer Festplatte des Service- und Wartungscomputers gespeichert sind. Schon die Auswahl des korrekten Programms erfordert
5 erhebliche Fachkenntnis.

Aus dem US-Patent 5,243,382 ist ein Steuersystem für einen Drucker oder Kopierer bekannt, bei dem ein tragbares Wartungsgerät an eine Wartungsschnittstelle anschließbar ist.
10 Der Drucker oder Kopierer überträgt erste Daten mit Zustandsinformationen des Druckers oder Kopierers mit Hilfe einer Verbindung zwischen Wartungsgerät und dem Drucker oder Kopierer. Weiterhin können dem Wartungsgerät zweite
15 Daten eingegeben werden, die Zustandsinformationen enthalten. Mindestens ein Paar von gespeicherten Steuerinformationen auf der Basis der ersten und zweiten Zustandsinformationen können durch das Wartungsgerät ausgegeben werden. Weiterhin ist aus dem US-Patent 5,243,382 bekannt, die
20 gespeicherten Daten zu einer Datenverarbeitungsanlage zu übertragen.

Aufgabe der Erfindung ist es, ein Verfahren und ein System zum Erzeugen einer grafischen Benutzeroberfläche für ein elektrofotografisches Druck- oder Kopiersystem anzugeben,
25 durch das eine an das jeweilige Druck- oder Kopiersystem angepasste Benutzeroberfläche einfach erzeugbar ist und zumindest eine Bedienfunktion und/oder Diagnosefunktion zum Bedienen bzw. zur Diagnose des Druck- oder Kopiersystems bereitgestellt wird.

30 Die Aufgabe wird durch ein Verfahren mit den Merkmalen des Patentanspruchs 1 gelöst. Vorteilhafte Weiterbildungen der Erfindung sind in den abhängigen Patentansprüchen angegeben.

35 Durch ein Verfahren mit den Merkmalen des Patentanspruchs 1 wird erreicht, dass eine grafische Benutzeroberfläche

zum Bedienen und/oder zur Diagnose des Druck- oder Kopiersystems auf einer Bedieneinheit erzeugt wird, ohne dass eine Installation von Spezialsoftware und eine Anpassung von Programmmodulen erforderlich ist. Eine druck- oder kopiersystemspezifische Anpassung der Bedieneinheit kann
5 somit entfallen. Auch ist eine Auswahl eines für das Druck- oder Kopiersystem angepaßtes Programm aus einer Vielzahl von Programmen ist nicht erforderlich.

10 Die zum Erzeugen der grafischen Benutzeroberfläche erforderlichen Daten sind im Druck- oder Kopiersystem selbst gespeichert. Diese Daten sind bereits an den Typ und den Ausgabestand des Druck- oder Kopiersystems bzw. von Baueinheiten des Druck- oder Kopiersystems angepasst, wodurch
15 eine korrekte Bedienung, korrekte Konfiguration und/oder eine korrekte Diagnose des Druck- oder Kopiersystems sehr einfach möglich ist.

Ein zweiter Aspekt der Erfindung betrifft ein System zum
20 Erzeugen einer grafischen Benutzeroberfläche für ein elektrofotografisches Druck- oder Kopiersystem. Eine erste Datenverarbeitungseinheit des Druck- oder Kopiersystems enthält einen ersten Speicherbereich, in dem erste Daten zum Erzeugen einer grafischen Benutzeroberfläche gespeichert
25 sind. Die ersten Daten werden zu einer zweiten Datenverarbeitungseinheit einer Bedieneinheit übertragen, wobei die zweite Datenverarbeitungseinheit über eine Datenleitung mit der ersten Datenverarbeitungseinheit verbunden ist. Die zweite Datenverarbeitungseinheit verarbeitet
30 die ersten Daten mit Hilfe eines Anzeigeprogrammmoduls, das eine grafische Benutzeroberfläche erzeugt. In einem zweiten Speicherbereich der ersten Datenverarbeitungseinheit sind zweite Daten gespeichert, die Programmelemente enthalten. Die zweiten Daten werden über die Datenleitung zur zweiten Datenverarbeitungseinheit übertragen.
35 Die zweite Datenverarbeitungseinheit verarbeitet die zweiten Daten und stellt zumindest eine Bedien- und/oder

Diagnosefunktion zum Bedienen bzw. zur Diagnose des Druck- oder Kopiersystems bereit.

5 Durch dieses erfindungsgemäße System wird erreicht, dass die Bedieneinheit keine speziell an das Druck- oder Kopiersystem angepasste Programmelemente und Programmmodule benötigt, um eine für das jeweilige Druck- oder Kopiersystem angepasste Benutzeroberfläche bzw. angepasste Bedienoberflächen auszugeben. Sowohl die Daten zum Erzeugen
10 einer grafischen Benutzeroberfläche als auch Programmelemente zum Bedienen und/oder zur Diagnose des Druck- oder Kopiersystems sind in einem Speicherbereich des Druck- oder Kopiersystems gespeichert, wobei sowohl die grafische Benutzeroberfläche als auch die Programmelemente auf den
15 Typ und/oder den Ausgabestand des Druck- oder Kopiersystems abgestimmt sind.

Ein dritter Aspekt der Erfindung betrifft ein Verfahren zum Laden von Programmdateien zum Bedienen und/oder zur
20 Diagnose eines elektrofotografischen Druckers oder Kopierers. Erste Daten werden in einem ersten Speicherbereich einer ersten Datenverarbeitungseinheit eines Druckers oder Kopierers gespeichert. Die ersten Daten werden von der ersten Datenverarbeitungseinheit zu einer zweiten Daten-
25 verarbeitungseinheit einer Bedieneinheit übertragen. Die ersten Daten enthalten zumindest Angaben über mindestens ein zum Erzeugen von Bedien- und/oder Diagnosefunktionen erforderliches Programmmodul. Mit Hilfe der zweiten Datenverarbeitungseinheit wird überprüft, ob zweite Daten, die
30 Programmmodul enthalten, in einem zweiten Speicherbereich der zweiten Datenverarbeitungseinheit enthalten sind. Bei nicht vorhandenen zweiten Daten im zweiten Speicherbereich werden die zweiten Daten aus einem dritten Speicherbereich der ersten Datenverarbeitungseinheit zur
35 zweiten Datenverarbeitungseinheit übertragen. Anweisungen des Programmmoduls werden von der zweiten Datenverarbeitungseinheit abgearbeitet.

Durch dieses erfindungsgemäße Verfahren zum Laden von Programm-
daten wird erreicht, dass zumindest die zweiten Daten
in einer Bedieneinheit gespeichert werden können, wobei
5 mit Hilfe der ersten Daten einfach überprüft werden kann,
ob im zweiten Speicherbereich das benötigte Programmmodul
enthalten ist. Nur dann, wenn das erforderliche Pro-
grammmodul nicht im zweiten Speicherbereich enthalten ist,
werden die zweiten Daten, die das Programmmodul enthalten,
10 von der ersten Datenverarbeitungseinheit zur zweiten Da-
tenverarbeitungseinheit übertragen. Die erforderliche
Übertragungszeit insbesondere zum Laden der zweiten Daten
wird dadurch wesentlich verkürzt, vor allem dann, wenn zur
Datenübertragung Datenleitungen mit einer geringen Daten-
15 Übertragungsrate genutzt werden.

Ein vierter Aspekt der Erfindung betrifft ein System zum
Laden von Programmdaten zum Bedienen und/oder zur Diagnose
eines elektrofotografischen Druckers oder Kopierers. Das
20 System enthält eine erste Datenverarbeitungseinheit des
Druckers, die mit einer zweiten Datenverarbeitungseinheit
einer Bedieneinheit über eine Datenleitung verbunden ist.
Von der ersten Datenverarbeitungseinheit zur zweiten Da-
tenverarbeitungseinheit sind erste Daten übertragbar, die
25 zumindest Angaben über mindestens ein zum Erzeugen von
Bedien- und/oder Diagnosefunktionen erforderliches Pro-
grammmodul enthalten. Die zweite Datenverarbeitungseinheit
überprüft, ob zweite Daten, die das Programmmodul enthal-
ten, in einem zweiten Speicherbereich der zweiten Daten-
30 verarbeitungseinheit gespeichert sind. Bei nicht vorhande-
nen zweiten Daten im zweiten Speicherbereich werden die
zweiten Daten von einem dritten Speicherbereich der ersten
Datenverarbeitungseinheit zur zweiten Datenverarbeitungs-
einheit übertragen. Die zweite Datenverarbeitungseinheit
35 führt Anweisungen des Programmmoduls aus. Dadurch wird
erreicht, daß die zweiten Daten nicht vom Drucker oder
Kopiersystem übertragen werden müssen, wenn sie bereits in

einem Speicherbereich der Bedieneinheit gespeichert sind. Wartezeiten zum Laden der Daten sind verringert.

5 Zum besseren Verständnis der vorliegenden Erfindung wird
im Folgenden auf die in den Zeichnungen dargestellten
bevorzugten Ausführungsbeispiele Bezug genommen, die an
Hand spezifischer Terminologie beschrieben sind. Es sei
jedoch darauf hingewiesen, dass der Schutzzumfang der Er-
findung dadurch nicht eingeschränkt werden soll, da derar-
10 tige Veränderungen und weitere Modifizierungen an den
gezeigten Vorrichtungen und/oder den Verfahren sowie der-
artige weitere Anwendungen der Erfindung, wie sie darin
aufgezeigt sind, als übliches derzeitiges oder künftiges
Fachwissen eines zuständigen Fachmannes angesehen werden.
15 Die Figuren zeigen Ausführungsbeispiele der Erfindung,
nämlich:

Figur 1 ein Blockschaltbild eines Systems mit einem Drucker und einem Service- und Wartungscomputer gemäß einem ersten Ausführungsbeispiel der Erfindung;
20

Figur 2 ein Blockschaltbild mit einer Detailansicht einer Ein- und Ausgabesteuerung des Druckers und des Service- und Wartungscomputers nach Figur 1;
25

Figur 3 ein Blockschaltbild, das die Datenübertragung und Kommunikationsstruktur zwischen dem Service- und Wartungscomputer und dem Drucker nach Figur 1 zeigt;
30

Figur 4 eine Anordnung eines Druckers und eines Service- und Wartungscomputers gemäß einem zweiten Ausführungsbeispiel der Erfindung;
35

Figur 5 eine Anordnung des Druckers und des Service- und Wartungscomputers nach Figur 4, wobei der Ser-

vice- und Wartungscomputer über eine Datenfern-
Übertragungsverbindung mit dem Drucker verbunden
ist;

5 Figur 6 ein Blockschaltbild, in dem der Zugriff eines
Anzeigeprogrammmoduls auf Programmelemente dar-
gestellt ist;

10 Figur 7 ein Blockschaltbild mit einer Datenverarbei-
tungseinheit eines Service- und Wartungscompu-
ters und einer Druckersteuerung, wobei die Da-
tenübertragung zum Übertragen von Anzeigedaten
dargestellt ist;

15 Figur 8 einen Service- und Wartungscomputer, der mit
einem ersten und einem fünften Drucker verbunden
ist;

20 Figur 9 einen Service- und Wartungscomputer, der mit
einem zweiten Drucker verbunden ist; und

25 Figur 10 Programmdateienarchive zum Erzeugen einer grafi-
schen Benutzeroberfläche zur Konfiguration und
Diagnose.

30 In Figur 1 ist ein Blockschaltbild eines Systems 10 zur
Wartung, Konfiguration und Diagnose eines Druckers 12 mit
Hilfe eines Service- und Wartungscomputers 18 dargestellt.
Der Drucker 12 enthält eine Druckersteuerung 14 und eine
35 Ein- und Ausgabesteuerung 16. Die Gerätesteuerung 14 ent-
hält alle zur Steuerung, Wartung, Konfiguration und Feh-
leranalyse sowie zur Diagnose des Druckers 12 erforderli-
chen Informationen und Messwerte. Die Gerätesteuerung 14
ist mit der Ein- und Ausgabesteuerung 16 verbunden. Über
eine erste Netzwerkverbindung ist ein erster Service- und
Wartungscomputer 18 mit dem Drucker 12 verbindbar. Der
Service- und Wartungscomputer 18 ist z.B. ein Service-

Notebook, das ein Servicetechniker bei Serviceeinsätzen vor Ort am Drucker 12, z.B. bei einem Kunden, mit der Ein- und Ausgabesteuerung 16 des Druckers 12 verbindet. Die Verbindung erfolgt vorzugsweise mit Hilfe einer Netzwerkverbindung eines Local Area Network (LAN). Dazu verbindet der Servicetechniker das Service-Notebook 18 mit Hilfe eines Netzkabels, eines sogenannten Patch-Kabels, mit einem ersten Netzwerkanschluss 20 der Ein- und Ausgabesteuerung 16. Weiterhin enthält die Ein- und Ausgabesteuerung 16 einen zweiten Netzwerkanschluss 22, durch den der Drucker 12 mit einem Netzwerk 24 verbunden ist. Über das Netzwerk 24 ist ein externes Bedienfeld 26 und ein LAN-Modem 28 mit dem Drucker 12 verbunden. Das LAN-Modem 28 ist mit einem Telefonnetz 30 verbunden, wobei über das LAN-Modem 28 eine Datenverbindung zu einem Servicezentrum des Druckerherstellers und zu einem Telefonanschluss eines Servicetechnikers herstellbar ist. In Figur 1 ist der Telefonanschluss des Servicetechnikers und das Servicezentrum, die jeweils mit dem Telefonnetz 30 verbunden sind, nicht dargestellt.

Mit Hilfe des externen Bedienfeldes 26 können Bedienpersonen Bedieneingaben durchführen. So können die Bedienpersonen z.B. Papierparameter eingeben, Druckbilder positionieren, Druckqualitäten festlegen und das Abarbeiten von Druckaufträgen koordinieren. Mit Hilfe des Service-Notebooks 18 können sowohl die Bedienhandlungen durchgeführt werden, die Bedienpersonen über das externe Bedienfeld 26 durchführen können, als auch Einstellwerte und Parameter des Druckers 12 verändert und Speicherbereiche und Register der Gerätesteuerung 14 ausgegeben und mit neuen Daten beschrieben werden. In gleicher Weise, wie mit Hilfe des Service-Notebooks 18, stehen einem Servicetechniker beim Anschluss eines weiteren Service-Notebooks am Telefonanschluss und an einem mit dem Telefonnetz über in Modem verbundenen Service- und Wartungscomputer in einer Serviceleitstelle des Druckerherstellers jeweils Funktionen

zur Diagnose und zur Einstellung von Druckerparametern zur Verfügung. Über das Telefonnetz 30 ist somit eine Fernwartung, Fernkonfiguration und Ferndiagnose des Druckers 12 möglich und mit Hilfe des mit dem Drucker 12 verbundenen Service-Notebooks 18 eine lokale Diagnose und Wartung.

In Figur 2 ist ein Ausschnitt des Blockschaltbildes nach Figur 1 mit dem Service-Notebook 18 und der Ein- und Ausgabesteuerung 16 dargestellt. Gleiche Elemente haben gleiche Bezugszeichen. Das Service-Notebook 18 enthält neben einem Betriebssystem ein Anzeigeprogrammmodul 32, das auch als Browser-Programmmodul bezeichnet wird. Das Browser-Programmmodul 32 enthält eine sogenannte Java-Runtime-Umgebung, durch die es möglich ist, Programmelemente, die in der Programmiersprache Java erstellt worden sind, abzuarbeiten, wobei die durch diese Programmelemente erzeugten Daten mit Hilfe des Browser-Programmmoduls 32 ausgegeben werden.

Die Ein- und Ausgabesteuerung 16 enthält einen sogenannten HTTP-Server 36, der mindestens einen ersten Datenspeicherbereich 38 enthält, in dem Hypertexte und Elemente zum Erzeugen sogenannter Hypertext-Mark-up-Language-Seiten (HTML-Seiten) gespeichert sind, wobei die gespeicherten Daten vorzugsweise sogenannte Hypertexte zur Seitenbeschreibung enthalten. Mit Hilfe dieser HTML-Seiten wird eine grafische Benutzeroberfläche erzeugt, bei der, vorzugsweise über ein Menü, mehrere Bedienoberflächen auswählbar sind. Die Bedienoberflächen werden in einem in der grafischen Benutzeroberfläche vorgesehenen Abschnitt ausgegeben.

Weiterhin enthält der HTTP-Server 36 einen Speicherbereich 40, in dem Programmelemente gespeichert sind, die in der Programmiersprache Java erstellt worden sind. Ein solches Programmelement wird auch als Java-Applet bezeichnet. Die Programmdateien des Java-Applets, sowie Klassendaten, die

zum Abarbeiten des Java-Applets benötigt werden, und Programm-
 daten von Programmteilen, die durch das Java-Applet
 aufgerufen werden, sind vorzugsweise in Archiven, soge-
 nannten jar-Archiven gespeichert. Vorzugsweise sind Daten
 5 n einem Archiv gespeichert, die zum Erzeugen einer Funkti-
 on oder einer Gruppe von Funktionen erforderlich sind.
 Ferner sind bei anderen Ausführungsformen Speicherbereich
 40 auch Java-Applikationen sowie Programmelemente gespei-
 chert, die als ActiveX-Programmelemente bezeichnet werden.
 10 Im Speicherbereich 40 sind vorzugsweise mehrere Archive
 mit Java-Applets bzw. ActiveX-Elemente enthalten. Vorzugs-
 weise ist ein Archiv mit Programm-
 daten zur Authentifizierung, ein Archiv mit Programm-
 daten zur Kommunikationssteuerung zwischen dem Service-Notebook 18 und der Ein-
 15 und Ausgabesteuerung 16 sowie weitere Archive mit Pro-
 gramm-
 daten zum Bereitstellen von Bedien-, Konfigurations
 und Diagnosefunktionen enthalten.

Die Übertragung von Daten mit Einstellwerten und Drucker-
 20 parametern zwischen Drucker 12 und Service-Notebook 18
 erfolgt vorzugsweise mit Hilfe einer Remote Method Invoca-
 tion(RMI)-Kommunikation. Mit Hilfe einer solchen RMI-Kom-
 munikation ist es auch möglich, auf Objekte der Geräte-
 steuerung 14 und auf eine nicht dargestellte Datenbasis
 25 des Druckers 12, vorzugsweise auf eine Management Informa-
 tion Base (MIB) des Druckers 12, zuzugreifen. Ferner kön-
 nen im Speicherbereich 40 Programm-
 daten bzw. Archive mit
 Programm-
 daten zur Diagnose einer Papiereingabeeinheit, zur
 Diagnose einer Papierlaufsteuerung, zur Diagnose einer
 30 Foliarausgabeeinheit oder zur Diagnose einer Druckeinheit
 enthalten sein. Weiterhin sind Archive mit Programm-
 daten zum Zugriff auf eine Ereignisregistrierung und Archive mit
 Programm-
 daten zum Zugriff auf einen Fehlerspeicher im
 Speicherbereich 40 enthalten.

35

Über die Netzwerkverbindung zwischen dem Service-Notebook
 18 und dem Netzwerkanschluss 20 wird eine Verbindung mit

Hilfe eines Transmission Control Protocol/Internet Protocol (TCP/IP) aufgebaut. Durch eine Voreinstellung im Browser-Programmmodul 32 wird eine HTML-Seite vom HTTP-Server 36 angefordert. Die Daten der HTML-Seite werden aus dem Speicherbereich 38 ausgelesen und zum Browser-Programmmodul 32 übertragen. Durch Abarbeiten des übertragenen Hypertextes der HTML-Seite wird beim HTTP-Server 36 ein Java-Applet angefordert. Der HTTP-Server 36 liest Daten aus dem Speicherbereich 40 aus, die Programmdateien des Java-Applets enthalten. Die Programmdateien des Java-Applets werden zum Service-Notebook 18 übertragen und von der Java-Runtime-Programmu-
 5
 10

gebung 34 abgearbeitet.

Im einfachsten Fall enthält die HTML-Seite bzw. der übertragene Hypertext nur die Anweisung zum Nachladen eines oder mehrerer Java-Applets. Die grafische Benutzeroberfläche, Anzeigeelemente und Bedienelemente sowie die Bedien-, Konfigurations- und Diagnosefunktion werden dann, wie bereits erwähnt, mit Hilfe der Java-Applet-Programmelemente realisiert.

15
 20

Bei anderen Ausführungsbeispielen wird beim Aufbau der TCP/IP-Verbindung überprüft, ob der Servicetechniker mit dem Service-Notebook 18 autorisiert ist, Zugriff auf Daten des HTTP-Servers 36 zu erhalten. Das Java-Applet erzeugt die grafische Bedienoberfläche mit Hilfe des Browser-Programmmoduls 32 und baut mit Hilfe einer RMI-Kommunikation eine logische Verbindung zu einem RMI-Server 36 auf. Das Service-Notebook 18 ist bei der RMI-Kommunikation ein RMI-Client. Ferner sind die externe Bedieneinheit 26 und die über das Telefonnetz 30 angeschlossenen Servicecomputer RMI-Clients. Zum Übertragen von Massendaten, wie z.B. Fehlerdaten und sogenannten Trace-Daten, wird ein File-Transfer-Protokoll (FTP) zur Datenübertragung genutzt. Dazu ist in der Ein- und Ausgabesteuerung 16 ein spezieller FTP-Server vorgesehen, der jedoch nur zur Datenüber-

25
 30
 35

tragung aktiviert wird und sonst aus Datenschutzgründen nicht aktiv ist.

Zum Aktivieren des FTP-Servers werden mit Hilfe der RMI-Kommunikation Daten übertragen, durch die die Ein- und Ausgabesteuerung 16 ein sogenanntes SNMP-Kommando erzeugt (SNMP = Simple Network Management Protocol). Zum Übertragen von Anweisungen zu Steuereinheiten des Druckers 12, das Auslesen von Parametern aus dem Drucker 12, wie Sensorwerten, Fehlerständen und Statusinformationen, werden ebenfalls SNMP-Kommandos genutzt, die mit Hilfe der RMI-Kommunikation zwischen dem Service-Notebook 18 und dem Drucker 12 erzeugt werden. Mit Hilfe von SNMP-Kommandos kann auch auf in der bereits erwähnten Management Information Base (MIB) enthaltene Variablen, Parameter, Kommando-register und Statusregister zugegriffen werden, wobei jedem Element der Management Information Base ein Object Identifier (OID) als Adressierungsadresse zugeordnet ist. Die SNMP-Kommandos werden im RMI-Server mit Hilfe der übertragenen RMI-Kommandos und RMI-Daten erzeugt. Somit sind keine direkten Zugriffe mit SNMP-Kommandos von außen auf die MIB des Druckers 12 möglich. Dadurch wird sichergestellt, dass nicht berechnigte Personen und Steuereinheiten keinen Zugriff auf die Steuerungen 16 und die Datenbasis des Druckers 12 haben. Bei der Übertragung der Daten mit Hilfe des FTP-Protokolls werden die Daten jedoch direkt übertragen, wobei dann gesonderte Maßnahmen ergriffen werden, um einen unberechtigten Zugriff zu verhindern.

In Figur 3 ist ein Blockschaltbild dargestellt, in dem die RMI-/SNMP-Kommunikation zwischen dem Drucker 12 und dem Service-Notebook 18 dargestellt ist. Wie bereits im Zusammenhang mit Figur 2 erläutert, erfolgt die Kommunikation zwischen dem Service-Notebook 18 und dem Drucker 12 über eine Netzwerkverbindung mit Hilfe einer RMI-Kommunikation. Das Anzeigeprogramm 32 des Service-Notebooks 18 arbeitet ein Java-Applet 48 zum Erzeugen eines RMI-Clients einer

RMI-Kommunikation ab. Das Java-Applet 48 ermittelt aus den mit Hilfe der RMI-Kommunikation übertragenen Daten die SNMP-Daten. Die SNMP-Daten enthalten wiederum Daten, z.B. mit Variablenwerten, Messwerten und Einstellwerten des Druckers 12, denen jeweils eine OID-Adresse zugeordnet ist. Mit Hilfe der OID-Adresse werden die Daten einzelner Variablen in einer Management Information Base (MIB) adressiert. Durch eine Programmfunktion 46 wird diesen Daten dann abhängig von der OID-Adresse ein Variablenname zugeordnet. Mit Hilfe der Variablennamen werden die übertragenen Daten durch eine Programmfunktion 44 angefordert und weiterverarbeitet.

Die Programmfunktion 44 dient als SNMP-Manager zum Ausführen und zum Erzeugen von SNMP-Kommandos sowie zum Zwischenspeichern von Daten. Mit Hilfe einer Programmfunktion 42 zum Erzeugen einer grafischen Benutzeroberfläche werden die mit Hilfe der Programmfunktion 44 ermittelten Daten und SNMP-Kommandos in Anzeigedaten umgewandelt, die auf einer Anzeigeeinheit des Service-Notebooks 18 ausgegeben werden.

Weiterhin können über die grafische Benutzeroberfläche Bedieneingaben und Eingaben von Einstellwerten erfolgen, die dann mit Hilfe der Programmfunktion 44 in SNMP-Kommandos umgewandelt werden. Entsprechend den in diesen Kommandos enthaltenen Variablennamen werden durch die Programmfunktion 46 Variablenadressen, d.h. OID-Adressen, zugeordnet, wobei die Daten zum Erzeugen der SNMP-Kommandos zusammen mit den Variablenadressen mit Hilfe der RMI-Kommunikation zum Drucker 12 übertragen werden. Die Daten zum Erzeugen der SNMP-Kommandos enthalten insbesondere RMI-Kommandos. Die Ein- und Ausgabesteuerung 16 des Druckers 12 enthält ein Programmmodul zum Erzeugen von SNMP-Kommandos aus den mit Hilfe der RMI-Kommunikation übertragenen Daten. Die SNMP-Kommandos enthalten dabei auch Daten mit Einstellwerten. Die SNMP-Kommandos umfassen insbesondere ein Get-

Kommando zum Aufruf von Daten aus der Management Information Base, ein Set-Kommando zum Verändern von Daten in der Management Information Base und ein Trap-Kommando zur direkten Übertragung von Informationen zu einem SNMP-
 5 Agenten oder einem SNMP-Subagenten. Die SNMP-Agenten sind jeweils als Programmelemente in Steuereinheiten des Druckers 12, wie z.B. der Gerätesteuerung 14 enthalten. Die Programmdateien einer Programmfunktion 42, 44, 46 sind in jeweils einem Archiv enthalten und können so einzeln über-
 10 tragen, abgearbeitet und gespeichert werden.

Die Ein- und Ausgabesteuereinheit 16 überträgt die SNMP-Kommandos zu einem SNMP-Masteragenten 52, der mit Hilfe der dem jeweiligen Kommando zugeordneten Adresse zur be-
 15 treffenden Steuereinheit weiterleitet, in der ein Element mit dieser Adresse gespeichert ist. Dieses Weiterleiten des Kommandos zur jeweiligen Steuereinheit, von denen in Figur 3 die Steuereinheit 54 und die Steuereinheit 14 dargestellt sind, wird auch als Routen bezeichnet.

20 Vorzugsweise sind die SNMP-Agenten 52, 54, 14 sowie die mit Hilfe der SNMP-Agenten verwalteten Variablen und Daten derart hierarchisch organisiert, dass ihnen eine gemäß der Struktur der SNMP-Agenten entsprechende Adressstruktur
 25 zugeordnet ist, wodurch der Speicherort der jeweiligen Variable einfach mit Hilfe ihrer OID-Adresse einfach auffindbar ist. Mit Hilfe der SNMP-Kommandos können sowohl Diagnosefunktionen der einzelnen Steuereinheiten 14, 16, 52, 54 aufgerufen als auch Bedienhandlungen ausgeführt
 30 werden. Somit erfolgt die Datenübertragung zwischen der grafischen Benutzeroberfläche 42 und den Steuereinheiten 14, 16, 52, 54 des Druckers 12 mit Hilfe von SNMP-Kommandos, die zumindest zwischen dem Drucker 12 und dem Service-Notebook 18 mit Hilfe einer RMI-Kommunikation
 35 übertragen bzw. erzeugt werden.

In Figur 4 ist ein System 56 zur Wartung, Konfiguration, Diagnose und Bedienen eines Druckers 58 dargestellt. Der Drucker 58 enthält ein internes Bedienfeld 60, das mit einer Ein- und Ausgabesteuerung 62 des Druckers 58 verbunden ist. Weiterhin ist ein Service-Notebook 64 eines Servicetechnikers 66 über eine Local Area Network-Verbindung (LAN) 68 mit der Ein- und Ausgabesteuerung 62 des Druckers 58 verbunden.

10 In Figur 5 ist das System 56 nach Figur 4 dargestellt, wobei ein LAN-Modem 70 über eine LAN-Netzwerkverbindung 72 mit der Ein- und Ausgabesteuerung 62 verbunden ist. Mit Hilfe des LAN-Modems 70 wird über ein Telefonnetz 74 eine sogenannte Punkt-zu-Punkt-Verbindung zu einem zweiten LAN-Modem 76 aufgebaut. Die Telefonverbindung 74 ist wahlweise eine analoge Telefonverbindung oder eine ISDN-Telefonverbindung. Ein Service-Notebook 64 des Servicetechnikers 66 ist über eine Netzwerkverbindung 78 mit dem LAN-Modem 76 verbunden. Das LAN-Modem 76, das Service-Notebook 64 und der Servicetechniker 66 befinden sich in einer Serviceleitstelle 80 des Druckerherstellers. Der Servicetechniker 66 kann mit Hilfe des Service-Notebooks 64 die gleichen Bedien-, Konfigurations-, Diagnose- und Einstellhandlungen durchführen, die er bei einer direkten Netzwerkverbindung 68 zwischen dem Drucker 58 und dem Service-Notebook 64 nach Figur 4 durchführen kann. Jedoch ist bei einer Verbindung über das Telefonnetz 74 die Datenübertragungsrate wesentlich geringer als bei einer direkten Netzwerkverbindung 68 zwischen dem Service-Notebook 64 und dem Drucker 58.

In Figur 6 ist ein Blockschaltbild zum Verwalten und Auffinden von in jar-Archiven enthaltenen Daten dargestellt. In Zusammenhang mit den Figuren 2 bis 5 wurde bereits 35 erläutert, wie in jar-Archiven enthaltene Java-Applets und weitere Programmdateien mit Hilfe einer Java-Runtime-Umgebung des Browser-Programmmoduls 32 abgearbeitet

werden. Bei dem Ausführungsbeispiel nach Figur 2 werden die Java-Applets mit Hilfe von jar-Archiven vom Speicherbereich 40 zum Service-Notebook 18 übertragen. Bei der in Figur 6 dargestellten Anordnung sind jar-Archive in einem Archiv-Zwischenspeicherbereich 82 des Service-Notebooks 64 speicherbar. Die Anordnung nach Figur 6 enthält ein Browser-Programmmodul 84 zum Ausgeben einer grafischen Benutzeroberfläche 86. Zumindest Teile der grafischen Benutzeroberfläche 86 werden durch Ausführen eines Java-Applets 88 erzeugt.

Das Browser-Programmmodul 84 arbeitet einen Hypertext einer HTML-Seite ab, in der ein Programmaufruf eines Java-Applets 88 enthalten ist. Nach dem Aufruf des Java-Applets 88 wird es mit Hilfe der Java-Runtime-Programmumgebung 90 abgearbeitet, die auch als Java-Runtime-Environment bezeichnet wird. Zum Aarbeiten des Java-Applets 88 werden sogenannte Klassen aus mindestens einem jar-Archiv geladen. Das Nachladen erfolgt mit Hilfe sogenannter Classloader. Für den Fall, daß der Classloader die Klasse nicht in einem voreingestellten Archiv findet, wird die Anforderung zum Nachladen einem hierarchisch untergeordneten Classloader übergeben, der dann in einem weiteren voreingestellten Verzeichnis nach der angeforderten Klasse sucht. Die Java-Runtime-Programmumgebung enthält einen BootstrapClassloader 92 zum Laden von Klassen, die in einer Grundkonfiguration der Java-Runtime-Programmumgebung 90 enthalten sind. Ein ExtensionClassloader 94 dient zum Laden von Klassen, die den Funktionsumfang der Grundklassen erweitern und deren Zugriffsrechte durch ein in der Java-Runtime-Programmumgebung 90 enthaltenes Rechtekonzept nicht eingeschränkt ist. Ferner ist ein ApplicationClassloader 96 vorgesehen, der zum Laden von applet- bzw. applikations-spezifischen Klassen dient.

Die Klassen, die mit dem ApplicationClassloader 96 geladen werden, wurden speziell für die Applikation bzw. für das

Applet erstellt und sind nicht in einer Grundversion der Java-Runtime-Programmumgebung 90 mit enthalten. Der ApplicationClassLoader 96 dient zum Laden der Archivdaten und Quelldaten aller im Java-Applet 88 enthaltenen Quellen und Archive. Weiterhin ist ein CustomURLClassLoader 104 vorgesehen, der zusätzlich zu dem BootstrapClassLoader 92, dem ExtensionClassLoader 94 und dem ApplicationClassLoader 96 erfindungsgemäß vorgesehen ist. Der CustomURLClassLoader 104 dient zum Laden von applet- bzw. applikationsspezifischen Klassen und Programmdateien, die in einem voreinstellbaren Speicherbereich 82 gespeichert sind. Mit Hilfe des CustomURLClassloaders 104 ist es möglich, diese Klassen ohne die in den anderen Classloadern bestehenden Beschränkungen bei der Verwaltung der und dem Zugriff auf die Klassen aus beliebigen voreinstellbaren Speicherbereichen, z.B. aus Verzeichnissen 82, auszulesen.

Beim Zugriff auf eine vom CustomURLClassLoader 104 verwalteten Klasse wird eine Anforderung dieser Klasse zuerst einem sogenannten BootstrapClassLoader 92 zugeführt, der das voreingestellte Archiv nach der Klasse durchsucht. Ist die gesuchte Klasse nicht im voreingestellten Archiv enthalten, so wird anschließend ein weiteres Archiv mit einem ExtensionsClassLoader 94 nach der Klasse durchsucht. Ist die Klasse auch in diesem Archiv nicht enthalten, so wird nachfolgend ein Speicherbereich durch einen ApplicationClassLoader 96 nach der angeforderten Klasse durchsucht. Der ApplicationClassLoader 96 hat sowohl Zugriff auf einen Browser-Zwischenspeicher 98 und einen PlugIn-Zwischenspeicher 100.

Sowohl der Archiv-Zwischenspeicher 82, der Browser-Zwischenspeicher 98 und der PlugIn-Zwischenspeicher 100 sind in einem Festplattenspeicher einer Datenverarbeitungseinheit enthalten, wobei die Datenverarbeitungseinheit als Bedieneinheit und als Service- und Wartungscomputer des Druckers dient, mit dem eine Datenverarbeitungseinheit des

- Druckers über eine Netzwerkverbindung 102 verbunden ist. Der Service- und Wartungscomputer enthält auch das Browser-Programmmodul 84 mit der Java-Runtime-Programmumgebung 90. Sowohl im Browser-Zwischenspeicher 98 als auch im PlugIn-Zwischenspeicher 100 sind Daten, Klassen, Archive und Java-Applets zwischenspeicherbar, die von einem über das Netzwerk 102 angeschlossenen Drucker zum Service- und Wartungscomputer übertragen werden.
- Der ApplicationClassLoader 96 hat, wie bereits erwähnt, Zugriff auf den Browser-Zwischenspeicher 98 und den PlugIn-Zwischenspeicher 100. Jedoch werden die Daten, die vom Drucker über das Netzwerk 102 übertragen werden, im Browser-Zwischenspeicher 98 und im PlugIn-Zwischenspeicher 100 unter Bezug auf die Netzwerkadresse des Druckers gespeichert, von dem sie übertragen worden sind. Weiterhin werden die im Browser-Zwischenspeicher 98 und im PlugIn-Zwischenspeicher 100 gespeicherten Daten nach einem vor-eingestellten Zeitraum gelöscht. Bei einer Speicherung des aufgerufenen Java-Applets im Browser-Zwischenspeicher 98 oder im PlugIn-Zwischenspeicher 100 müssen identische Archive, die die angeforderte Klasse enthalten, bei einer Änderung der Netzwerkadresse des Druckers, z.B. bei einer Verbindung des Service- und Wartungscomputers über das Netzwerk 102 mit einem baugleichen zweiten Drucker, von diesem zweiten Drucker nochmals zum Service- und Wartungscomputer übertragen werden.
- Sowohl die Übertragung eines identischen Archivs als auch das Zwischenspeichern des Archivs unter Bezug auf eine zweite Netzwerkadresse im Browser-Zwischenspeicher 98 oder im PlugIn-Zwischenspeicher 100 ist zeitaufwendig und speicherplatzaufwendig. Insbesondere dann, wenn über das Netzwerk 102 nur eine Datenübertragung mit geringer Datenübertragungsrate möglich ist. Wenn z.B. zwischen dem Service- und Wartungscomputer und dem Drucker ein Telefonnetz, ähnlich dem Telefonnetz 74 nach Figur 5, vorgesehen ist,

kann die Datenübertragung mehrere Minuten und bei speicherintensiven Archiven bis zu mehreren Stunden dauern. Wird beim Durchsuchen der Zwischenspeicher 98 und 100 durch den ApplicationClassLoader 96 das Archiv mit der angeforderten Klasse nicht gefunden, so wird erfindungsgemäß ein CustomURLClassLoader 104 aufgerufen, dem der Archiv-Zwischenspeicher 82 zugeordnet ist.

In dem Archiv-Zwischenspeicher 82 werden die Programmdateien unabhängig von der Netzwerkadresse des Druckers gespeichert, von dem sie übertragen worden sind. Der Archiv-Zwischenspeicher 82 kann auch den Speicherbereich einer CD-ROM umfassen, auf der eine Vielzahl Archive verschiedenartiger Drucker und unterschiedlicher Versionen gleicher Druckertypen gespeichert sind. Alternativ oder zusätzlich sind auf der Festplatte eine Vielzahl von Archiven mit Klassen, Java-Applets und oder Java-Applikationen zum Bedienen und zur Diagnose verschiedener Drucker gespeichert. Beim Aufruf des CustomURLClassLoader 104 wird der Archiv-Zwischenspeicher 82 nach dem Archiv mit der durch das Browser-Programmmodul 84 aufgerufenen Klasse durchsucht. Ist das jeweilige Archiv im Archiv-Zwischenspeicher 82 gespeichert, so wird das Archiv aus diesem Archiv-Zwischenspeicher 82 ausgelesen und zur Java-Run-time-Programmumgebung 90 übertragen, die zusammen mit dem Browser-Programmmodul 84 das Archiv öffnet und die benötigten Daten, d.h. die angeforderten Klassendaten, ermittelt. Mit diesen Klassendaten kann das ausgeführte Java-Applet 88 benötigte Bedien- und/oder Diagnosefunktionen in der grafischen Benutzeroberfläche 86 zur Diagnose, Konfiguration, bzw. Bedienen des Druckers zur Verfügung stellen.

Ermittelt der CustomURLClassLoader 104 jedoch, dass kein Archiv mit den Klassendaten im Archiv-Zwischenspeicher 82 enthalten sind, so wird das Archiv vom über das Netzwerk 102 angeschlossenen Drucker angefordert und von diesem zur

Java-Runtime-Programmumgebung 90 des Service- und Wartungscomputers übertragen, wobei das Archiv nach dem Übertragen im Archiv-Zwischenspeicher 82 gespeichert wird. Bei einer erneuten Verbindung mit einem Drucker, bei dem das
 5 selbe Archiv benötigt wird, wird das Archiv dann aus dem Archiv-Zwischenspeicher 82 durch den CustomURLClassLoader 104 ausgelesen und eine angeforderte Klasse oder ein Java-Applet von der Java-Runtime-Programmumgebung 90 abgearbeitet. Ein erneutes Übertragen des Archivs vom angeschlossenen Drucker ist dann nicht erforderlich, auch dann
 10 nicht, wenn der Drucker eine andere Netzwerkadresse hat.

Die Archive enthalten vorzugsweise Programmdateien von Java-Applets oder von Teilen eines Java-Applets, Klassendaten
 15 oder andere Daten. Diese Archive enthalten eine Versionsinformation des Archivs und sind signiert, um eine Manipulation der Archivdaten und dadurch eine Manipulation des Druckers und der Bedieneinheit zu verhindern.

Werden bei einer vorhandenen Netzwerkverbindung 102 zwischen dem Drucker und dem Service- und Wartungscomputer keine Daten bzw. nur eine geringe Datenmenge übertragen, so kann der CustomURLClassLoader 104 weitere aktuell nicht benötigte Archive vom Drucker anfordern und im Archiv-Zwischenspeicher 82 speichern, die im Drucker gespeichert
 25 sind und im Archiv-Zwischenspeicher 82 noch nicht enthalten sind. Vorzugsweise werden Archive angefordert, deren Bezeichnungen in dem Hypertext der HTML-Seite 86 oder in einem bereits ausgeführten Java-Applet 88 enthalten sind.

Vorzugsweise wird zuerst ein jar-Archiv vom Drucker oder aus dem Archiv-Zwischenspeicher 82 vom Browser-Programmmodul 84 geladen und abgearbeitet, das einen Ur-Lader enthält, mit dem weitere Klassen und Programmdateien gezielt
 35 nachgeladen werden, die zum Erzeugen der grafischen Benutzeroberfläche 86 benötigt werden. Anschließend wird ein jar-Archiv mit einer Authentifizierungsprozedur zum Au-

thentifizieren der Datenverarbeitungsanlage durch den Drucker geladen. Nachfolgend wird ein jar-Archiv mit Java-Applets zur Kommunikation zwischen dem Drucker und dem Service- und Wartungscomputer geladen. Anschließend wird
 5 ein jar-Archiv mit Daten zum Erzeugen von Grundfunktionen der grafischen Benutzeroberfläche 86 geladen und ausgeführt.

Nach Aufruf einzelner Diagnose- und Bedienfunktionen über
 10 die grafische Benutzeroberfläche 86 werden dann zum Bereitstellen der Diagnose bzw. der Bedienfunktion erforderliche jar-Archive vom Archiv-Zwischenspeicher 82 oder vom Drucker nachgeladen. Vorzugsweise enthält der Hypertext der HTML-Seite oder ein jar-Archiv, das vom Drucker zum
 15 Service- und Wartungscomputer übertragen worden ist, die Bezeichnungen aller für mögliche Bedien- und Diagnosefunktionen erforderlichen jar-Archive. Alternativ sind diese Daten im Ur-Lader enthalten. Vorzugsweise enthält ein Java-Applet alle zum Bereitstellen der Bedien- bzw.
 20 Diagnosefunktion erforderlichen Programmschritte, um Abhängigkeiten zwischen einzelnen Archiven zu vermeiden. Dadurch wird ein zeitaufwendiges Nachladen weiterer jar-Archive vermieden, die an sich nicht oder noch nicht benötigt werden.

Bei anderen Ausführungsbeispielen werden Informationen über den Aufbau und Inhalt der grafischen Benutzeroberfläche 86 im Drucker gespeichert, insbesondere über den Aufbau eines Menübaums, den Inhalt von Listen und Ausgabefeldern. Diese gespeicherten Informationen werden beim erneuten Erzeugen einer grafischen Benutzeroberfläche 86 zur Beschleunigung des Aufbaus der grafischen Benutzeroberfläche 86 genutzt, indem vorerst nur die gespeicherten Informationen vom Drucker zu dem Service- und Wartungscomputer
 30 oder zu einer Bedieneinheit übertragen werden. So kann z.B. bei einem erneuten Serviceeinsatz auf diese im Drucker gespeicherten Informationen zurückgegriffen und eine
 35

grafische Benutzeroberfläche 86 mit den gespeicherten Informationen aufgebaut werden.

5 Auf das Laden einzelner jar-Archive aus dem Zwischenspeicher 82 und aus dem Drucker kann bis zu einem Zeitpunkt verzichtet werden, zu dem die jeweilige Funktion dann konkret benötigt wird, oder gegebenenfalls ganz, wenn die entsprechende Funktion bei der jeweiligen Bedienung und/oder Diagnose des Druckers nicht benötigt wird. Vorzugsweise werden beim Speichern der Informationen zusätzlich Versionsdaten gespeichert, die es ermöglichen, die Aktualität der gespeicherten Information zu ermitteln. Die Bedien-, Konfigurations- und Diagnosefunktionen betreffen insbesondere Parametereinstellungen des Druckers, das Ein-

10 stellen von Fehlergrenzwerten, das Einstellen von Spannungen zwischen einzelnen Bauteilen einer Bilderzeugungseinheit des Druckers, Statuseinstellungen des Druckers, Lichtschrankentestroutinen, Motorstarttroutinen und Ventiltestroutinen.

20

In Figur 7 ist in Blockschaltbild mit einer Datenverarbeitungseinheit 110 und einer Druckersteuerung 112 dargestellt, bei dem Elemente zur Datenübertragung zwischen der Druckersteuerung 112 und der Datenverarbeitungseinheit 110 dargestellt sind. Die Datenverarbeitungseinheit 110 ist beispielsweise ein Service-Notebook oder eine Bedieneinheit des Druckers. Die Druckersteuerung 112 enthält einen Web-Server 114, ähnlich dem HTTP-Server 36 nach Figur 2. Weiterhin enthält die Druckersteuerung 112 eine Firmware

25 116. Durch Abarbeiten der Firmware 116 durch die Druckersteuerung 112 wird zumindest ein Teil der Baueinheiten des Druckers gesteuert sowie ein RMI-Server zur RMI-Kommunikation zwischen Druckersteuerung 112 und Datenverarbeitungsanlage 110 erzeugt. Weiterhin werden mit Hilfe der Firmware

30 116 Messdaten und Einstelldaten des Druckers ermittelt. Die Messdaten und Einstelldaten sind in einem Speicherbereich 118 der Druckersteuerung 112 oder in Speicherberei-

35

chen nicht dargestellter Steuereinheiten der Druckersteuerung 112 enthalten. Die Firmware liest diese Messdaten aus dem Speicherbereich 118 und/oder aus den Steuereinheiten aus und stellt diese Daten mit Hilfe des RMI-Servers zum

5 Abruf durch die Datenverarbeitungsanlage 110 zur Verfügung. Mit Hilfe eines Browser-Programmmoduls 120 erzeugt die Datenverarbeitungseinheit 110 eine grafische Benutzeroberfläche, durch die Daten, insbesondere Meß- und Einstellwerte sowie Betriebszustände und Fehlerdaten anzeig-

10 bar sind.

Das Browser-Programmmodul 120 enthält eine mit Hilfe einer PlugIn-Programmmodul-Programmumgebung 122 bereitgestellte

15 Java-Runtime-Programmumgebung. Mit Hilfe der Programmumgebungen 122 führt das Browser-Programmmodul 120 Programmelemente aus, wie z.B. Java-Applets. Abzuarbeitende Daten, Klassendaten sowie die Programmdateien mehrerer Applets sind in Archiven in einem Speicherbereich 124 der Datenverarbeitungseinheit 110 enthalten. Der Speicherbereich 124

20 umfasst vorzugsweise einen Browser-Zwischenspeicher, einen PlugIn-Zwischenspeicher und einen Archiv-Zwischenspeicher, ähnlich den Zwischenspeichern nach Figur 6. Die Datenübertragung zwischen dem Web-Server 114 und dem Browser-Programmmodul 122 erfolgt vorzugsweise mit einem Hypertext-

25 Transferprotokoll. Das Übertragen von Messdaten und Einstellwerten des Druckers zu einem vom Browser-Programm 120 abgearbeiteten Java-Applet 126 erfolgt mit Hilfe einer RMI-Kommunikation zwischen dem Java-Applet 120 und dem RMI-Server, der durch die Abarbeitung der Firmware 116

30 erzeugt wird. Das Java-Applet 126 stellt Funktionen zur Konfiguration und Diagnose der Druckersteuerung 112 und des Druckers zur Verfügung, in dem die Druckersteuerung 112 enthalten ist.

35 In Figur 8 ist jeweils der Speicherinhalt des Browser-Zwischenspeichers 98, des PlugIn-Zwischenspeichers 100 und des Archiv-Zwischenspeichers 82 nach Figur 6 dargestellt.

Ein Service- und Wartungscomputer 130 ist mit einem ersten Drucker 132 und einem fünften Drucker 134 über jeweils eine Datenverbindung verbunden. Von den Druckern 132, 134 werden jeweils jar-Archive zum Service- und Wartungscomputer übertragen, um eine grafische Benutzeroberfläche sowie Bedien-, Konfigurations-, Diagnose- und Wartungsfunktionen bereitzustellen, wie in Zusammenhang mit Figur 6 und Figur 7 beschrieben. Vom Drucker 1 wird eine Datei "x.jar" und eine Datei "y1.jar" zum Service- und Wartungscomputer 130 übertragen. Der Service- und Wartungscomputer 130 ist so konfiguriert, dass die übertragenen Dateien sowohl im Browser-Zwischenspeicher, im PlugIn-Zwischenspeicher und im Archiv-Zwischenspeicher gespeichert werden. Im Browser-Zwischenspeicher und im PlugIn-Zwischenspeicher wird sowohl die Datei "x.jar" als auch die Datei "y1.jar" unter Angabe der Netzwerkadresse IP1 des Druckers 1 gespeichert. Im Archiv-Zwischenspeicher wird die Datei "x.jar" und die Datei "y1.jar" ohne Angabe einer Netzwerkadresse gespeichert. Mit Hilfe der Verbindung zwischen dem Drucker 134 und dem Service- und Wartungscomputer 130 wird die Datei "x.jar", "y5.jar" und "z.jar" zum Service- und Wartungscomputer 130 übertragen, wobei die Dateien "x.jar", "y5.jar" und "z.jar" unter Angabe der IP-Adresse IP5 im Browser-Zwischenspeicher und im PlugIn-Zwischenspeicher gespeichert werden. Die Datei "x.jar", die im Drucker 134 enthalten ist, und die Datei "x.jar", die im Drucker 152 enthalten ist, sind identisch. Im Archiv-Zwischenspeicher des Service- und Wartungscomputers 130 werden nur die Dateien "y5.jar" und "z.jar" ohne Angabe einer Netzwerkadresse gespeichert. Ist nur der Archiv-Zwischenspeicher als einziger Zwischenspeicher zum Zwischenspeichern der jar-Archive vorgesehen, so muss die Datei "x.jar" nur einmal entweder vom Drucker 132 oder vom Drucker 134 zum Service- und Wartungscomputer 130 übertragen werden.

In Figur 9 ist ein Blockschaltbild dargestellt, in dem der Service- und Wartungscomputer 130 nach Figur 8 mit einem

zweiten Drucker 136 verbunden ist. Vor dem Verbinden des Service- und Wartungscomputers 130 mit dem zweiten Drucker 136 ist der Service- und Wartungscomputer 130 bereits mit dem ersten und fünften Drucker 132, 134 verbunden worden, wie in Zusammenhang mit Figur 8 bereits erläutert. Mit Hilfe der Verbindung zwischen dem Service- und Wartungscomputer 130 und dem zweiten Drucker 136 werden die Programmmodule "x.jar" und "y2.jar" vom Drucker 136 zum Service- und Wartungscomputer 130 übertragen und unter Bezug auf die IP-Adresse des Druckers 136 im Browser-Zwischenspeicher und im PlugIn-Zwischenspeicher gespeichert. Wird, wie bereits in Zusammenhang mit Figur 8 erläutert, jedoch anstatt des Browser-Zwischenspeichers und des PlugIn-Zwischenspeichers erfindungsgemäß ein Archiv-Zwischenspeicher vorgesehen, in dem die jar-Archive gespeichert werden, so wird vom Drucker 136 nur das im Archiv-Zwischenspeicher nicht enthaltene jar-Archiv "y2.jar" vom Drucker 2 zum Service- und Wartungscomputer 130 übertragen.

Nach dem Verbinden des Service- und Wartungscomputers 130 mit den Druckern 132, 134 und 136 sind im Browser-Zwischenspeicher und im PlugIn-Zwischenspeicher jeweils sieben Dateien mit jar-Archiven gespeichert, die von den Druckern 132, 134, 136 zum Service- und Wartungscomputer 130 übertragen worden sind. Im Archiv-Zwischenspeicher sind nur fünf Dateien zwischengespeichert, da die Dateien "x.jar", die in den Druckern 132, 134, 136 gespeichert sind, übereinstimmen und nicht mehrfach von den Druckern 132, 134, 136 zum Service- und Wartungscomputer 130 übertragen werden müssen.

Je nach Voreinstellung im Browser-Programmmodul werden die im Browser-Zwischenspeicher und im PlugIn-Zwischenspeicher gespeicherten Daten nach einem vorbestimmten Zeitraum gelöscht. Die im Archiv-Zwischenspeicher enthaltenen Daten und Dateien werden zumindest nicht automatisch gelöscht.

Wird nur der Archiv-Zwischenspeicher des Service- und Wartungscomputers 130 zum Speichern der jar-Archive genutzt, so kann die zu übertragende Datenmenge von den Druckern 132, 134, 136 zu dem Service- und Wartungscomputer 130 somit erheblich verringert werden, wodurch Wartezeiten vermieden werden. Durch eine Überprüfung der erforderlichen Versionen der jar-Archive und der im Archiv-Zwischenspeicher gespeicherten Versionen der jar-Archive ist sichergestellt, dass die erforderlichen jar-Archive geladen und anschließend vom Browser-Programmmodul des Service- und Wartungscomputers 130 verarbeitet werden.

In Figur 10 ist ein Blockschaltbild mit jar-Archiven zum Erzeugen einer grafischen Benutzeroberfläche zur Konfiguration und Diagnose eines elektrofotografischen Druckers dargestellt. Die jar-Archive enthalten, wie bereits erwähnt, Java-Applets und/oder Java-Applikationen sowie zum Abarbeiten von Applets und Applikationen benötigte weitere Programmdateien. Ein erstes jar-Archiv 140 enthält mindestens ein Java-Applet, durch das eine Ur-Lader-Funktion bereitgestellt wird. Ein zweites jar-Archiv 142 enthält zumindest Klassen- oder Programmdateien zum Verwalten von Daten des Druckers im Browser-Programmmodul. Ein drittes jar-Archiv 144 enthält zumindest Klassen- oder Programmdateien zum Bereitstellen von Kommunikationsfunktionen, wodurch eine sogenannte Kommunikationsschicht erzeugt wird. Die von der Kommunikationsschicht benötigten Funktionen werden durch die im jar-Archiv 142 enthaltenen Klassen- oder Programmdateien bereitgestellt.

Weiterhin ist ein jar-Archiv 146 vorgesehen, das zumindest Klassen- oder Programmdateien zum Erzeugen allgemeiner Menükomponenten und der grafischen Strukturierung der grafischen Benutzeroberfläche enthält. Diese allgemeinen Menükomponenten greifen auf Funktionen der Kommunikationsschicht zurück. Weiterhin sind jar-Archive 148, 150, 152 und 154 vorgesehen, die jeweils eine Menügruppe und Funk-

tionen einzelner aufrufbarer Menüpunkte mit Hilfe von Klassen- und Programmdateien bereitstellen. Die Menügruppen 148 bis 154 greifen auf Funktionen der allgemeinen Menükomponenten 146 zurück. Weiterhin greift die Menügruppe 5 152 auf die Kommunikationsschicht 144 direkt zu. Die Menügruppen 148 bis 154 haben untereinander keine Abhängigkeit.

Nach dem Verbinden des Service- und Wartungscomputers mit dem Drucker wird das jar-Archiv 140 übertragen, das eine 10 relativ geringe Datenmenge enthält. Das jar-Archiv 140 enthält, wie bereits erwähnt, den Ur-Lader und weiterhin druckerspezifische Daten. Die weiteren zum Erzeugen der grafischen Benutzeroberfläche erforderlichen jar-Archive 15 142 bis 154 werden vorzugsweise aus dem Archiv-Zwischenspeicher 82 geladen. Der Archiv-Zwischenspeicher 82 befindet sich auf einem Festplattenspeicher des Service- und Wartungscomputers. Dadurch können diese Daten sehr schnell zum Browser-Programmmodul übertragen werden.

Anschließend wird das jar-Archiv 142 "ManagerFrame" zum Verwalten von Daten und das jar-Archiv 144 zur Kommunikation sowie zur Authentifizierung geladen. Erst nach erfolgter Authentifizierung werden weitere jar-Archive 146 25 bis 154 geladen. Beim Erzeugen der allgemeinen Menükomponenten mit Hilfe des jar-Archivs 146 werden Informationen genutzt, die im Drucker gespeichert sind und auf Einstellungen bereits konfigurierter Benutzeroberfläche basieren. Die jar-Archive 148 bis 154 werden erst zum Browser-Programmmodul übertragen, nachdem eine in der jeweiligen 30 Menügruppe enthaltene Funktion über die grafische Benutzeroberfläche angefordert worden ist. Ferner können weitere jar-Archive vom Drucker zum Service- und Wartungscomputer übertragen werden, die aktuell nicht vom Browser-Programmmodul zum Erzeugen der grafischen Benutzeroberfläche 35 erforderlich sind. Vorzugsweise erfolgt dieses Übertragen der Daten zu Zeitpunkten, in denen keine weiteren

Daten vom Drucker zum Service- und Wartungscomputer zu übertragen sind. Vorzugsweise werden alle vom Drucker zum Service- und Wartungscomputer übertragenen jar-Archive in dem Archiv-Zwischenspeicher 82 gespeichert, wenn die Archive noch nicht im Archiv-Zwischenspeicher 82 enthalten sind oder nur in einer anderen Programmversion im Archiv-Zwischenspeicher 82 enthalten sind.

Ferner können vom Drucker Daten zum Service- und Wartungscomputer übertragen werden, die Java-Applikationen enthalten, wobei die Abarbeitung dieser Java-Applikationen mit Hilfe einer sogenannten Java-Web-Start-Technologie erfolgt. Bei der Java-Web-Start-Technologie erfolgt eine Ausführung des Java-Applets oder einer Java-Applikation nicht mit Hilfe eines Browser-Programmmoduls sondern durch Abarbeiten einer Java-Network-Launching-Protokoll-Datei. Dadurch wird das vom Web-Server geladene Applet bzw. die geladene Applikation als lokale Applikation auf der Datenverarbeitungsanlage der Bedieneinheit bzw. des Service-Notebooks abgearbeitet. Als Datenleitungen zum Übertragen von Daten werden insbesondere drahtgebundene und drahtlose Datenverbindungen, wie Bluetooth, UMTS und DSL eingesetzt.

Obgleich in den Zeichnungen und in der vorhergehenden Beschreibung bevorzugte Ausführungsbeispiele aufgezeigt und detailliert beschrieben sind, sollte dies als rein beispielhaft und die Erfindung nicht einschränkend angesehen werden. Es wird darauf hingewiesen, dass nur die bevorzugten Ausführungsbeispiele dargestellt und beschrieben sind. Alle sämtlichen Veränderungen und Modifizierungen, die derzeit und künftig im Schutzzumfang der Erfindung liegen, geschützt werden sollen.

Bezugszeichenliste

	10	System
	12	Drucker
5	14	Gerätesteuerung
	16	Ein- und Ausgabesteuerung
	18	Service- und Wartungscomputer
	20, 22	Netzwerkanschluss
	24	Netzwerk
10	26	externe Bedieneinheit
	28	LAN-Modem
	30	Telefonnetz
	32	Anzeigeprogramm
	34	Java-Runtime-Programmumgebung
15	36	HTTP-Server
	38, 40	Speicherbereich
	42, 44, 46, 48	Programmfunktionen
	50	RMI-Server
	52, 54	SNMP-Agent
20	56	System
	58	Drucker
	60	Bedieneinheit
	62	Ein- und Ausgabesteuereinheit
	64	Service- und Wartungscomputer
25	66	Servicetechniker
	68, 72, 78	Netzwerkverbindung
	70, 76	LAN-Modem
	74	Telefonnetz
	80	Serviceleitstelle
30	82, 98, 100	Zwischenspeicher
	84	Browser-Programmmodul
	86	grafische Benutzeroberfläche
	88	Java-Applet
	90	Java-Runtime-Programmumgebung
35	92	BootstrapClassLoader
	94	ExtensionsClassLoader
	96	ApplicationClassLoader

	102	Netzwerk zum Drucker
	104	CustomURLClassLoader
	110	Datenverarbeitungseinheit
	112	Druckersteuerung
5	114	Web-Server
	116	Firmware
	118	Datenspeicher
	120	Browser-Programmmodul
	122	Java-Runtime-Programmumgebung
10	124	Datenspeicher
	126	Java-Applet
	130	Service- und Wartungscomputer
	132, 134, 136	Drucker
	140 bis 154	jar-Archive
15		

Ansprüche

1. Verfahren zum Erzeugen einer grafischen Benutzeroberfläche für ein elektrofotografisches Druck- oder Kopiersystem,
- bei dem in einem Speicherbereich (38) einer ersten Datenverarbeitungseinheit (16, 36) des Druck- oder Kopiersystems (12) erste Daten zum Erzeugen einer grafischen Benutzeroberfläche gespeichert werden,
- die ersten Daten zu einer zweiten Datenverarbeitungseinheit (18, 26) einer Bedieneinheit übertragen werden, die über eine Datenleitung mit der ersten Datenverarbeitungseinheit (16, 36) verbunden ist,
- die ersten Daten durch die zweite Datenverarbeitungseinheit (18, 26) verarbeitet werden,
- die zweite Datenverarbeitungseinheit (18, 26) ein Anzeigeprogrammmodul (32) abarbeitet, das die ersten Daten verarbeitet,
- in einem zweiten Speicherbereich (40) der ersten Datenverarbeitungseinheit zweite Daten gespeichert sind, die zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden,
- und bei dem die zweiten Daten von der zweiten Datenverarbeitungseinheit (18, 26) verarbeitet werden, wobei zumindest eine Bedienfunktion und/oder Diagnosefunktion zum Bedienen bzw. zur Diagnose des Druck- oder Kopiersystems (12) bereitgestellt wird.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die ersten Daten zumindest Seitenbeschreibungs-

formationen zum Erzeugen der grafischen Benutzeroberfläche, insbesondere einen Hypertext, enthalten.

- 5 3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die ersten Daten grafische Elemente zum Erzeugen der grafischen Benutzeroberfläche enthalten.
- 10 4. Verfahren nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, dass die ersten Daten mit Hilfe einer Hypertext-Mark-up-Language und/oder mit Hilfe der Programmiersprache Java erzeugt worden sind.
- 15 5. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass das Anzeigeprogrammmodul (32) ein Browser-Programmmodul zum Anzeigen einer grafischen Benutzeroberfläche mit Hilfe von in Hypertexten enthaltenen Daten und/oder von in Programmdateien zum Erzeugen einer grafischen Benutzeroberfläche enthaltenen Daten ist.
- 20 6. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die in den zweiten Daten enthaltenen Programmelemente in Archiven gespeichert sind, wobei die Programmelemente Klassendaten, Java-Applets und/oder ActiveX-Programmelemente umfassen, die vom Anzeigeprogrammmodul verarbeitet werden, wodurch Funktionen der grafischen Benutzeroberfläche, zum Bedienen, zur Konfiguration und/oder zur Diagnose des Druck- oder Kopiersystems (12) realisiert werden.
- 25 7. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Datenverbindung zwischen der ersten Datenverarbeitungseinheit (16, 36) und der zweiten Datenverarbeitungseinheit (18, 26) eine Netzwerkverbindung (24) ist.
- 30 8. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Datenverbindung zwischen der ersten Datenverarbeitungseinheit (16, 36) und der zweiten Datenverarbeitungseinheit (18, 26) eine Netzwerkverbindung (24) ist.
- 35 9. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Datenverbindung zwischen der ersten Datenverarbeitungseinheit (16, 36) und der zweiten Datenverarbeitungseinheit (18, 26) eine Netzwerkverbindung (24) ist.

8. Verfahren nach Anspruch 7, dadurch gekennzeichnet, dass die Netzwerkverbindung (24) mit Hilfe eines Local Area Networks erfolgt.
- 5 9. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Bedieneinheit ein Service- und Wartungscomputer zur Diagnose, Wartung und Parametereinstellung des Druck- oder Kopiersystems (12) ist.
- 10 10. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die ersten Daten und/oder die zweiten Daten mit Hilfe eines Hypertext-Transferprotokolls von der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden, und dass der erste und/oder zweite Speicherbereich (38, 40) einem HTTP-Server (36) der ersten Datenverarbeitungseinheit (16) zugeordnet ist, und dass bei einer Bedienhandlung zum Aufruf der Bedien-, Konfigurations- und/oder Diagnosefunktion die zweiten Daten von der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden.
- 15 11. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass das Anzeigeprogramm (32) eine Java-Runtime-Programmumgebung (34) enthält.
- 20 12. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass zumindest die zweiten Daten mit Hilfe einer Remote Method Invocation-Kommunikation übertragen werden.
- 25 13. Verfahren nach Anspruch 12, dadurch gekennzeichnet, dass mit Hilfe der Remote Methode Invocation-Kommunikation Anweisungen eines Simple Network Management-Protokolls übertragen werden.
- 30
- 35

14. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass dritte Daten von der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden.
15. Verfahren nach Anspruch 16, dadurch gekennzeichnet, dass die dritten Daten mit Hilfe eines File Transfer-Protokolls übertragen werden.
16. Verfahren nach einem der Ansprüche 14 oder 15, dadurch gekennzeichnet, dass die dritten Daten Fehlerdaten und Diagnoseprogramme enthalten.
17. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die zweite Datenverarbeitungseinheit (18, 26) über ein Wide Area Network mit der ersten Datenverarbeitungseinheit (16) verbunden ist.
18. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass vor dem Übertragen der ersten und/oder zweiten Daten überprüft wird, ob die zweite Datenverarbeitungseinheit (18, 26) berechtigt ist, die Daten zu empfangen und/oder zu senden, und dass die Authentizität der zweiten Datenverarbeitungseinheit (18, 26) mit Hilfe einer Authentifizierungsprozedur durch die erste und/oder zweite Datenverarbeitungseinheit (16, 18) überprüft wird.
19. System zum Erzeugen einer grafischen Benutzeroberfläche für ein elektrofotografisches Druck- oder Kopiersystem
- mit einer ersten Datenverarbeitungseinheit (16, 36) des Druck- oder Kopiersystems (12), die einen ersten Speicherbereich (38) enthält, in dem erste Daten zum

Erzeugen einer grafischen Benutzeroberfläche gespeichert sind,

5 bei dem die ersten Daten zu einer zweiten Datenverarbeitungseinheit (18, 26) einer Bedieneinheit übertragbar sind, wobei die zweite Datenverarbeitungseinheit (18, 26) über eine Datenleitung mit der ersten Datenverarbeitungseinheit (16, 36) verbunden ist,

10 die zweite Datenverarbeitungseinheit (18, 26) die ersten Daten mit Hilfe eines Anzeigeprogrammoduls (32) verarbeitet,

15 in einem zweiten Speicherbereich (40) der ersten Datenverarbeitungseinheit (16, 36) zweite Daten gespeichert sind, die Programmelemente enthalten, und die über die Datenleitung zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden,

20 und bei dem die zweite Datenverarbeitungseinheit (18, 26) die zweiten Daten verarbeitet und zumindest eine Bedien- und/oder Diagnosefunktion zum Bedienen bzw. zur Diagnose des Druck- oder Kopiersystems (12) bereitstellt.

25 20. Verfahren zum Laden von Programmdateien zum Bedienen und/oder zur Diagnose eines elektrofotografischen Druckers oder Kopierers,

30 bei dem erste Daten in einem ersten Speicherbereich (38) einer ersten Datenverarbeitungseinheit (16) eines Druckers (12) oder Kopierers gespeichert werden,

35 die ersten Daten von der ersten Datenverarbeitungseinheit (16) zu einer zweiten Datenverarbeitungseinheit (18, 26) einer Bedieneinheit übertragen werden, wobei die ersten Daten zumindest Angaben über mindestens ein

zum Erzeugen von Bedien- und/oder Diagnosefunktionen erforderliches Programmmodul enthalten,

5 mit Hilfe der zweiten Datenverarbeitungseinheit (18, 26) überprüft wird, ob zweite Daten, die das Programmmodul enthalten, in einem zweiten Speicherbereich (86) der zweiten Datenverarbeitungseinheit (18, 26) enthalten sind,

10 bei nicht vorhandenen zweiten Daten im zweiten Speicherbereich (86) die zweiten Daten aus einem dritten Speicherbereich (40) der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragen werden,

15

und bei dem Anweisungen des Programmmoduls durch die zweite Datenverarbeitungseinheit (18, 26) abgearbeitet werden.

20 21. Verfahren nach Anspruch 20, dadurch gekennzeichnet, dass der zweite Speicherbereich (86) ein Speicherbereich eines Festplattenspeichers und/oder eines austauschbaren Datenträgers ist.

25 22. Verfahren nach Anspruch 20 oder 21, dadurch gekennzeichnet, dass die ersten Daten zumindest den Drucker- typ und/oder den Ausgabestand des Druckers (12) oder Kopierers enthalten.

30 23. Verfahren nach einem der Ansprüche 20 bis 22, dadurch gekennzeichnet, dass im zweiten Speicherbereich (86) mehrere Programmmodule gespeichert sind, wobei mit Hilfe der ersten Daten ein Programmmodul ausgewählt wird.

35

24. Verfahren nach einem der Ansprüche 20 bis 23, dadurch gekennzeichnet, dass die ersten Daten ein Programmmodul-

dul enthalten, bei dessen Abarbeitung weitere erste Daten und zweite Daten geladen werden, wobei die ersten Daten ein Programmmodul zur Authentifizierung, ein Programmmodul zur Kommunikationssteuerung zwischen
5 ersten und zweiter Datenverarbeitungseinheit, ein Programmmodul zum Bereitstellen von Bedien- und/oder Diagnosefunktionen enthalten, wobei die ersten Daten in mindestens einer Datei enthalten sind.

10 25. Verfahren nach Anspruch 24, dadurch gekennzeichnet, dass die ersten und/oder zweiten Daten ein Java-Applet oder ein ActiveX-Programmelement enthalten.

15 26. Verfahren nach einem der Ansprüche 20 bis 25, dadurch gekennzeichnet, dass die Bedieneinheit (18, 26) ein Service- und Wartungscomputer ist.

20 27. Verfahren nach einem der Ansprüche 20 bis 26, dadurch gekennzeichnet, dass die erste Datenverarbeitungseinheit (16) über eine Datenfernübertragungs-Verbindung (28, 30) mit der zweiten Datenverarbeitungseinheit (18) verbunden ist.

25 28. Verfahren nach Anspruch 27, dadurch gekennzeichnet, dass die Datenfernübertragungs-Verbindung eine Punkt-zu-Punkt-Verbindung, insbesondere eine ISDN-Datenverbindung, eine Verbindung mit Hilfe eines Local Area Networks oder eine Verbindung mit Hilfe eines Wide Area Networks ist.

30 29. Verfahren nach einem der Ansprüche 20 bis 28, dadurch gekennzeichnet, dass die ersten und/oder zweiten Daten ein Ur-Lader-Programm, druckerspezifische Informationen, Programmmodule für eine RMI-Kommunikation, Programmmodule zum Durchführen einer Authentifizierung,
35 Programmmodule zum Erzeugen einer grafischen Benutzeroberfläche, Programmmodule zum Zugriff auf eine Da-

tenbasis des Druckers oder Kopierers, Programmmodule zur Diagnose einer Papiereingabeeinheit, einer Papierlaufsteuerung, einer Papierausgabeeinheit oder einer Druckeinheit, Programmmodule zum Zugriff auf eine Ereignisregistrierung und/oder Programmmodule zum Zugriff auf einen Fehlerspeicher enthalten.

30. Verfahren nach einem der Ansprüche 20 bis 29, dadurch gekennzeichnet, dass die ersten und/oder zweiten Daten Java-Applikationen enthalten, die mit Hilfe einer Java-Web-Start-Technologie zur zweiten Datenverarbeitungseinheit (18, 26) übertragen und durch diese abgearbeitet werden.

31. Verfahren nach einem der Ansprüche 20 bis 30, dadurch gekennzeichnet, dass die zweiten Daten Programmelemente zum Einstellen von Parametern, Zählerwerten, Zählergrenzwerten, einzustellenden Spannungswerten, Statusinformationen sowie zum Durchführen von Lichtschrankentestroutinen, Motortestroutinen und Ventiltestroutinen enthalten.

32. System zum Bedienen und/oder zur Diagnose eines elektrofotografischen Druckers oder Kopierers

mit einer ersten Datenverarbeitungseinheit (16) des Druckers (12) oder Kopierers, die mit einer zweiten Datenverarbeitungseinheit (18, 26) einer Bedieneinheit über eine Datenleitung verbunden ist,

bei dem erste Daten von der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragbar sind, die zumindest Angaben über mindestens ein zum Erzeugen von Bedien- und/oder Diagnosefunktionen erforderliches Programmmodul enthalten,

die zweite Datenverarbeitungseinheit (18, 26) überprüft, ob zweite Daten, die das Programmmodul enthalten, in einem zweiten Speicherbereich (86) der zweiten Datenverarbeitungseinheit (18, 26) gespeichert sind,

5

bei nicht vorhandenen zweiten Daten im zweiten Speicherbereich (86) die zweiten Daten von einem dritten Speicherbereich der ersten Datenverarbeitungseinheit (16) zur zweiten Datenverarbeitungseinheit (18, 26) übertragbar sind,

10

und bei dem die zweite Datenverarbeitungseinheit (18, 26) Anweisungen des Programmmoduls ausführt.

- 15 33. System nach Anspruch 32, dadurch gekennzeichnet, dass die zweiten Daten nach dem Übertragen im zweiten Speicherbereich (86) speicherbar sind.

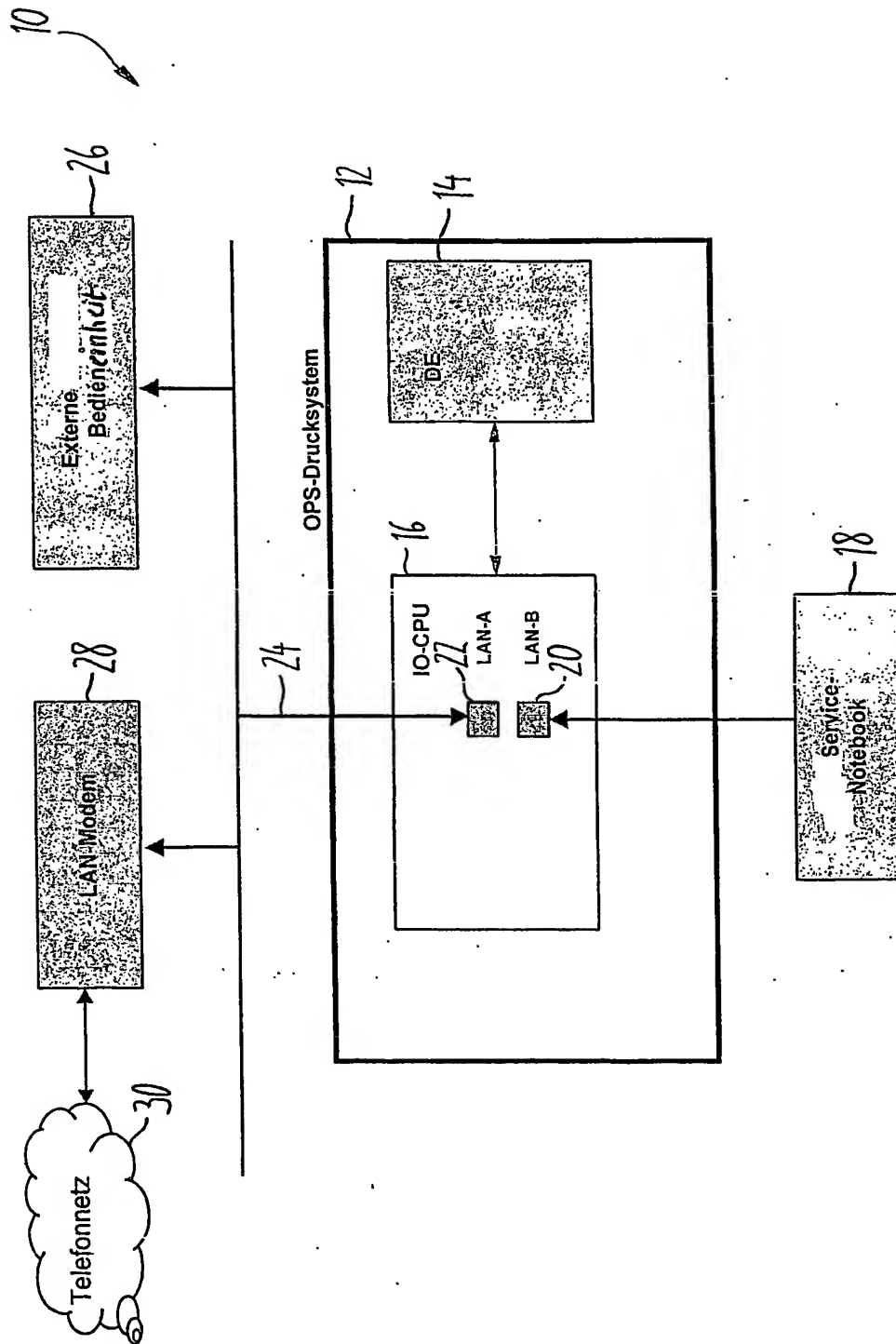
Zusammenfassung

Verfahren und System zum Erzeugen einer grafischen Benut-
zeroberfläche zum Bedienen, zum Konfigurieren und/oder zur
5 Diagnose eines elektrofotografischen Druckers oder Kopie-
rers

Die Erfindung betrifft ein Verfahren und ein System zum
Erzeugen einer grafischen Benutzeroberfläche für ein elek-
10 trofotografisches Druck- oder Kopiersystem. In einem er-
sten Speicherbereich (38) einer ersten Datenverarbeitungse-
inheit (16) des Druck- oder Kopiersystems werden erste
Daten zum Erzeugen einer grafischen Benutzeroberfläche
gespeichert. Die ersten Daten werden zu einer zweiten
15 Datenverarbeitungseinheit (18, 26) einer Bedieneinheit
übertragen, die über eine Datenleitung (24) mit der ersten
Datenverarbeitungseinheit (16) verbunden ist. Die ersten
Daten werden durch die zweite Datenverarbeitungseinheit
(18, 26) verarbeitet. Die zweite Datenverarbeitungseinheit
20 (18, 26) arbeitet ein Anzeigeprogrammmodul (32) ab, das
die ersten Daten verarbeitet. In einem zweiten Speicherbe-
reich (40) der ersten Datenverarbeitungseinheit (16) sind
zweite Daten gespeichert, die zur zweiten Datenverarbei-
tungseinheit (18, 26) übertragen werden. Die zweiten Daten
25 werden von der zweiten Datenverarbeitungseinheit (18, 26)
verarbeitet, wobei zumindest eine Bedienfunktion und/oder
Diagnosefunktion zum Bedienen bzw. zur Diagnose des Druck-
oder Kopiersystems (12) bereitgestellt wird.

30 Figur 1)

ZUSAMMENFASSUNG



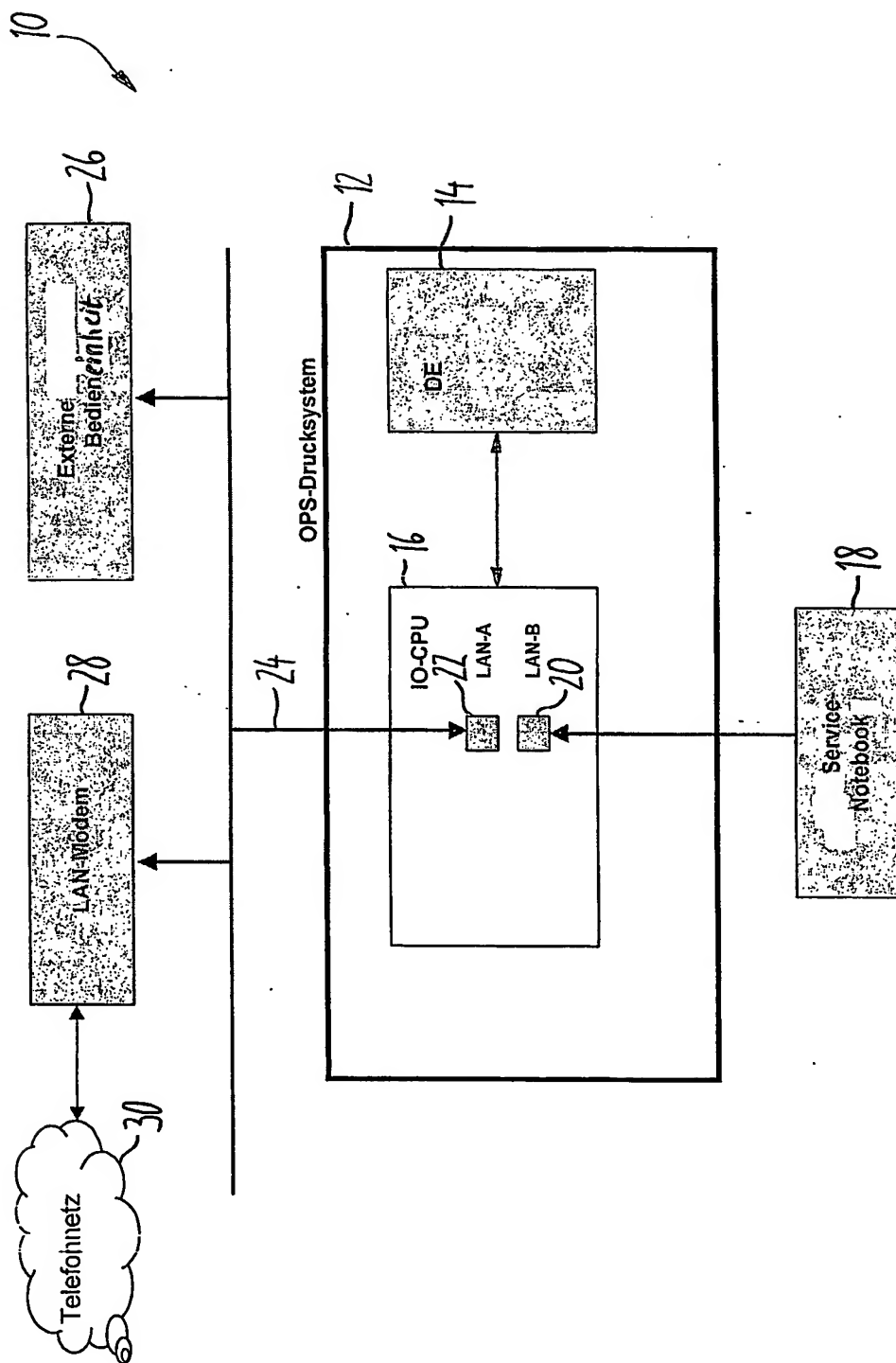
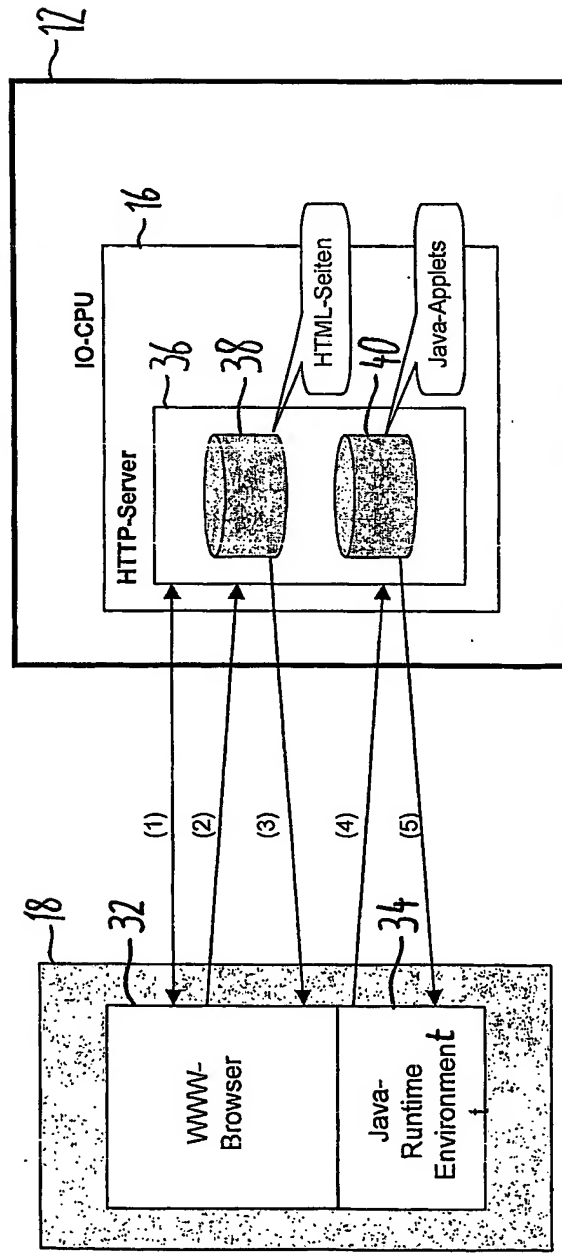


Fig. 1



- (1) Aufbau TCP/IP Verbindung
- (2) Anforderung HTML-Seite
- (3) Übertragung HTML-Seite
- (4) Anforderung Java-Applet
- (5) Übertragung Java-Applet

Fig. 2

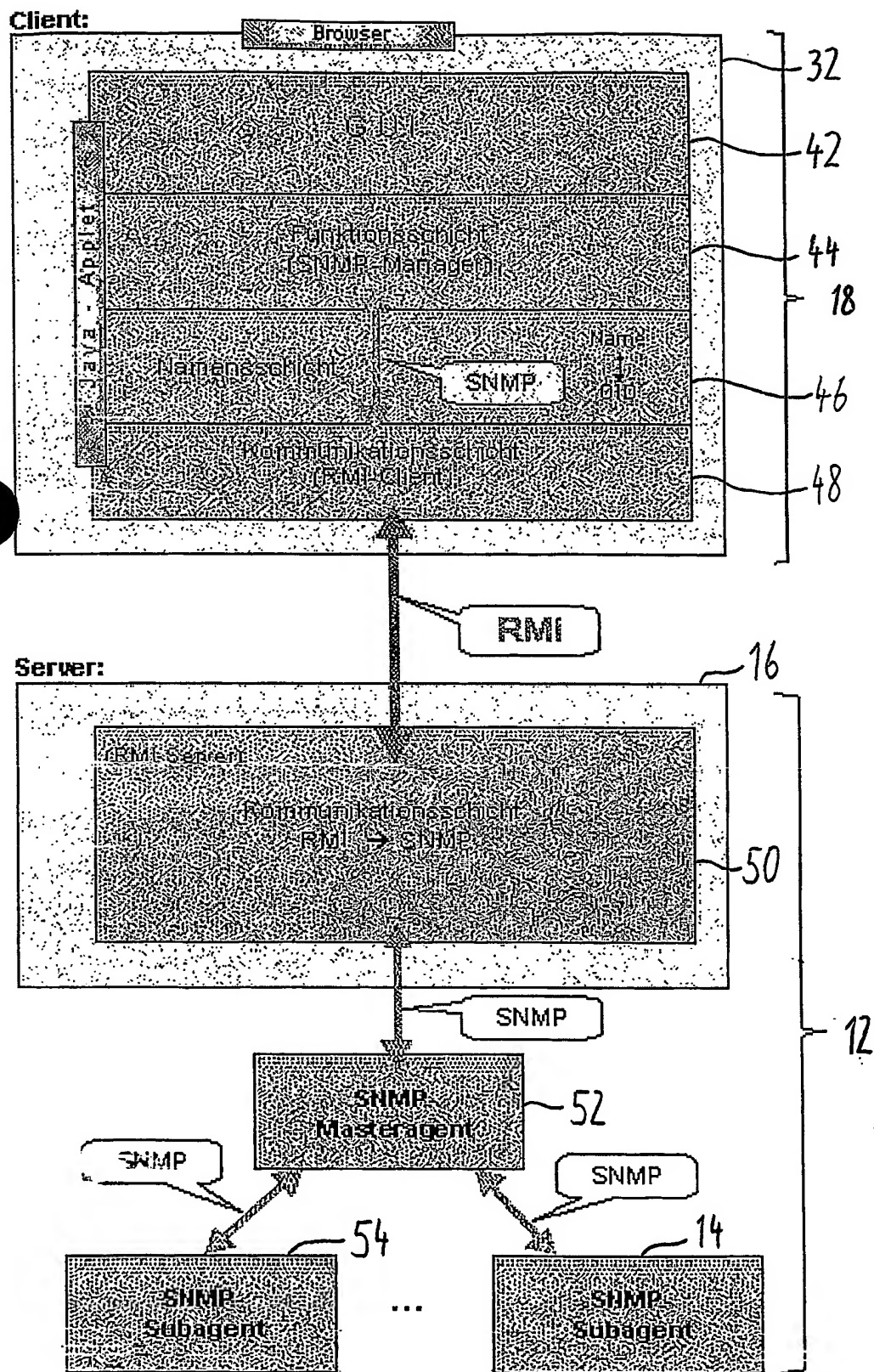


Fig. 3

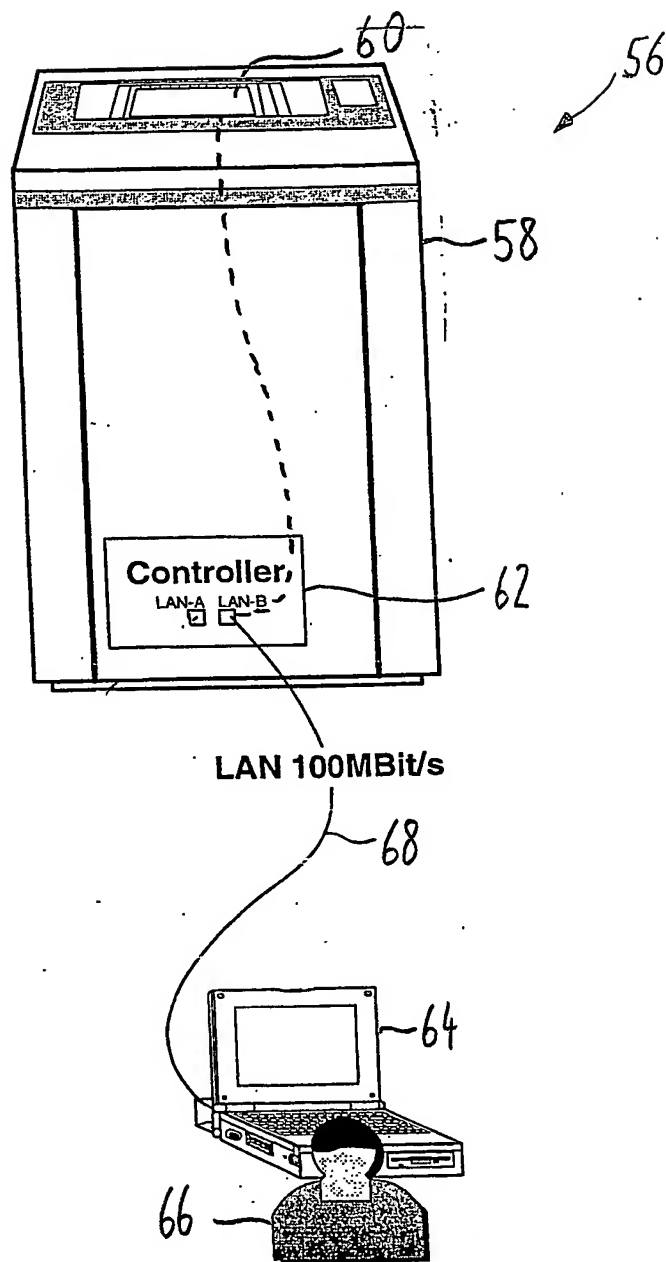


Fig. 4

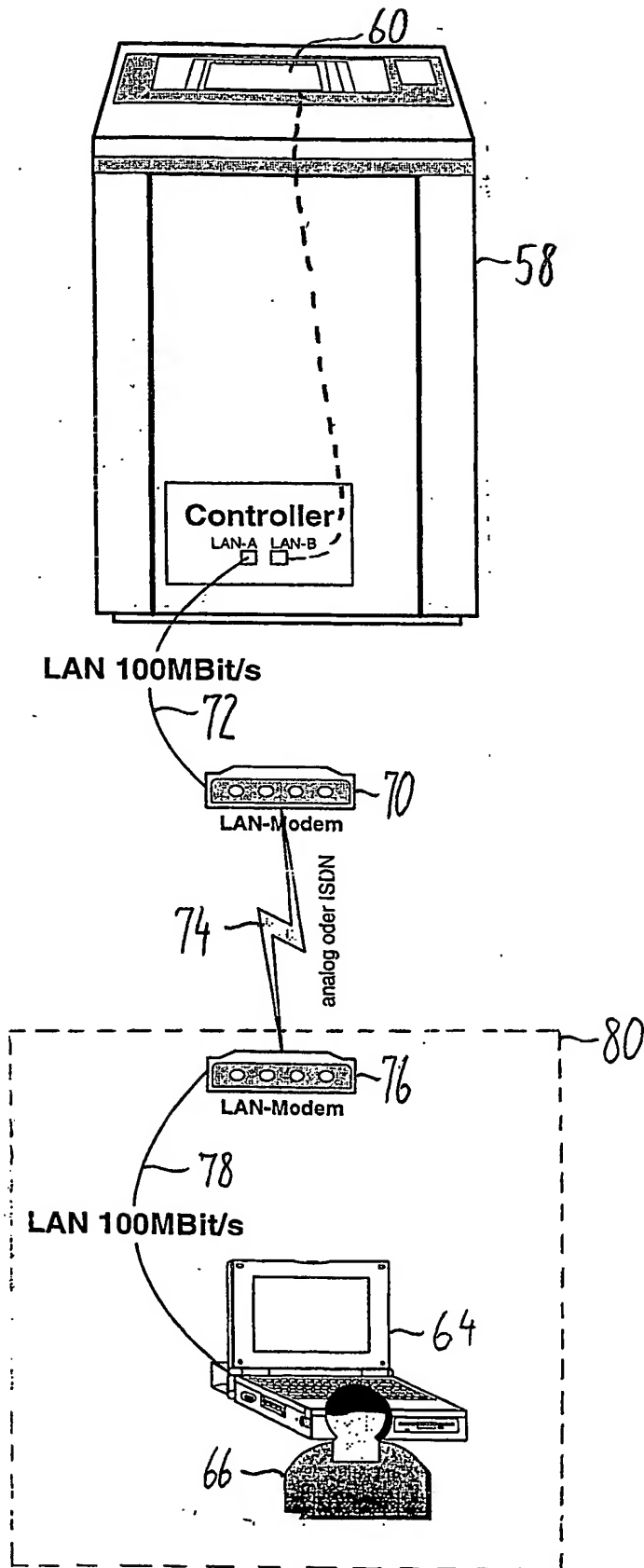


Fig. 5

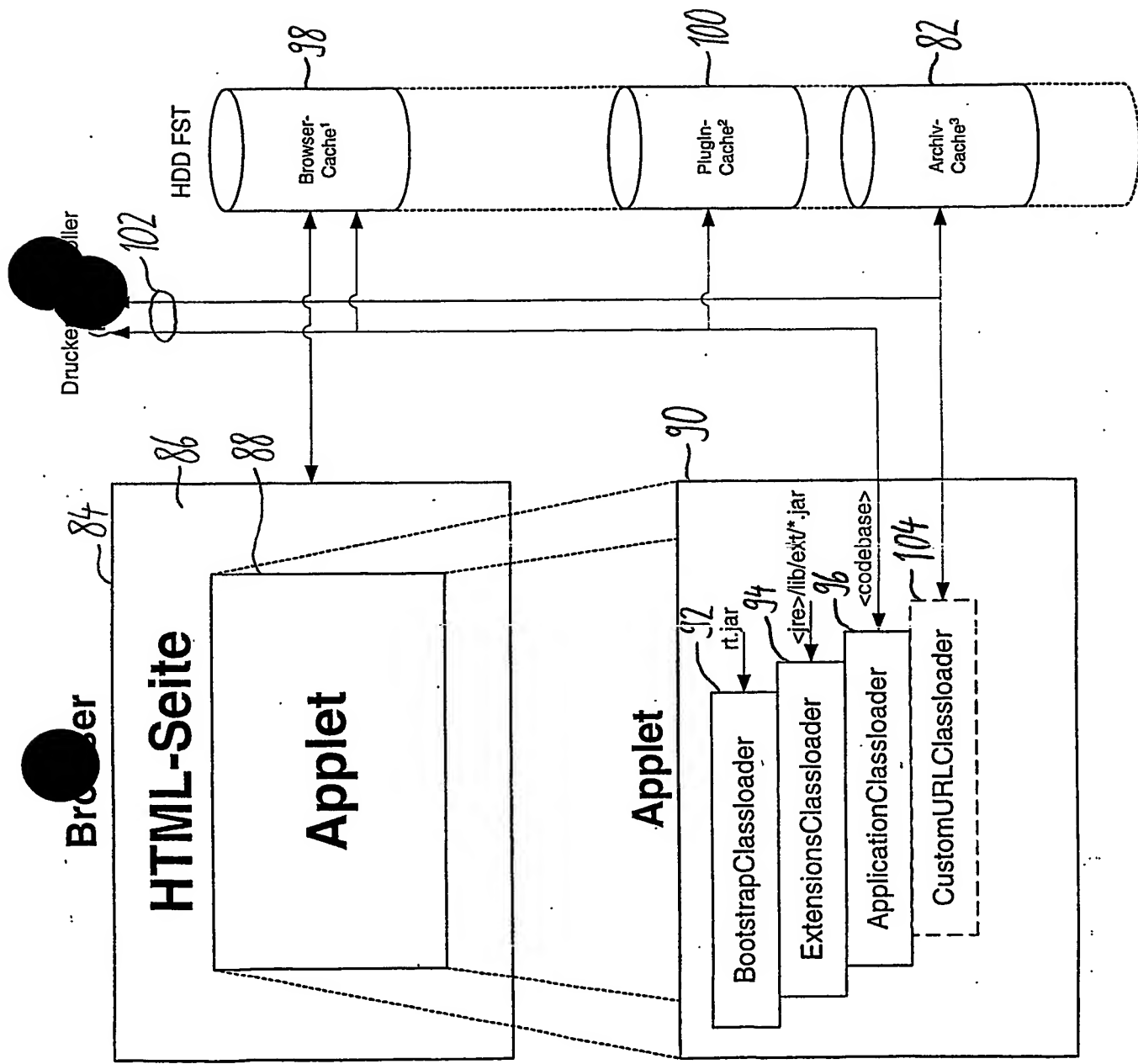


Fig. 6

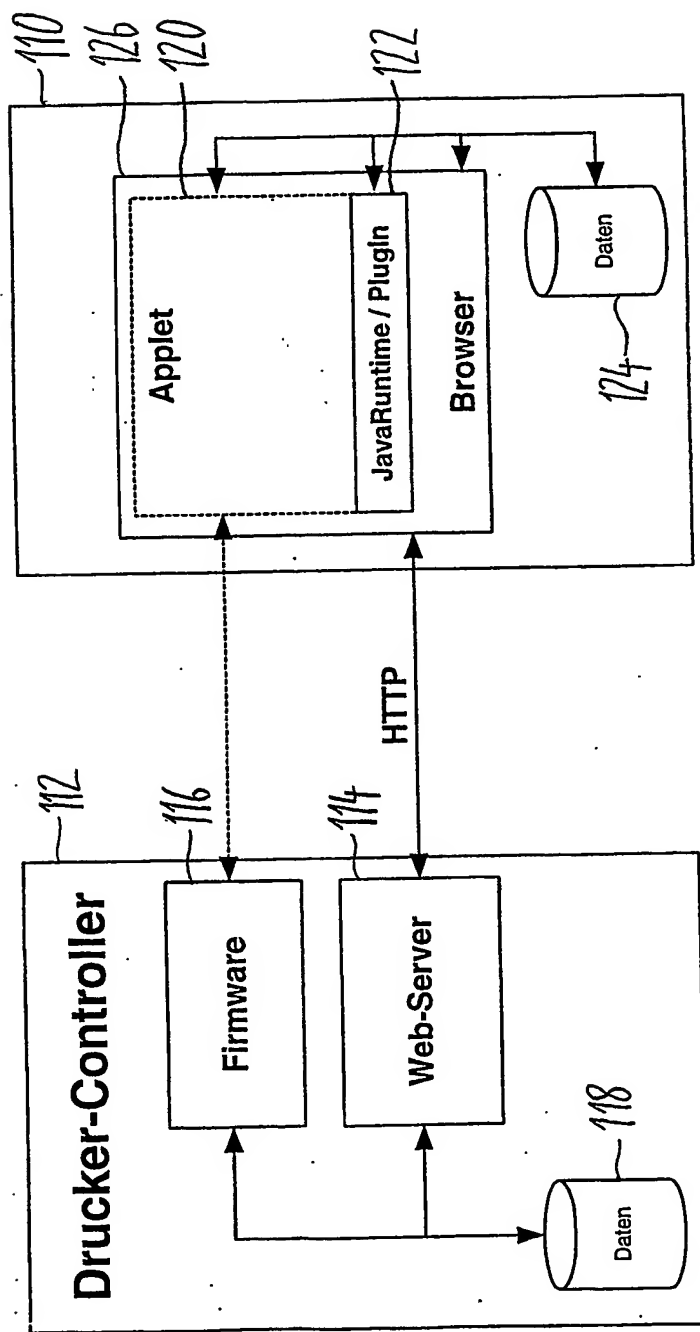


Fig. 7

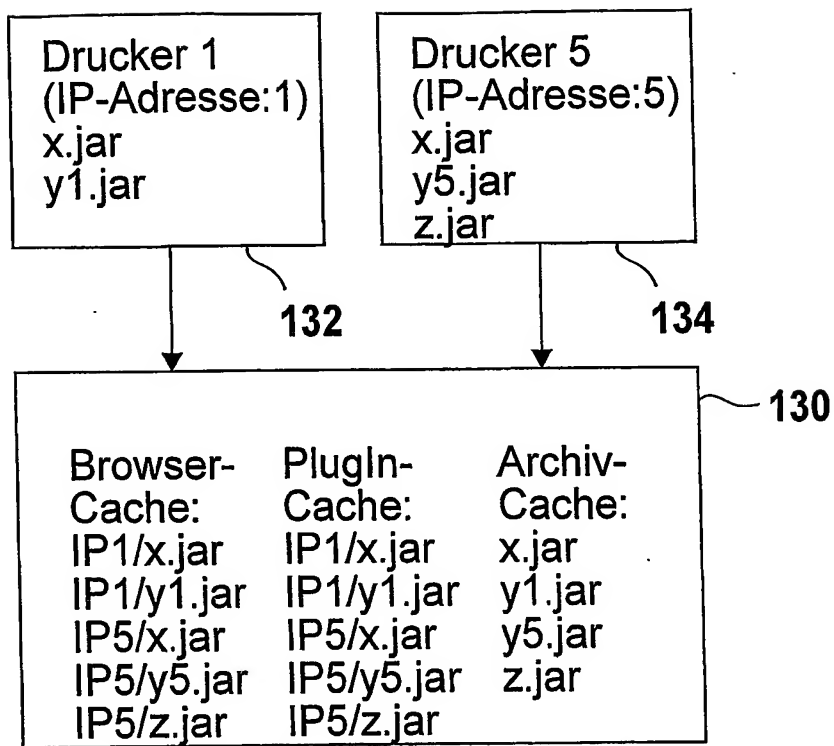


Fig. 8

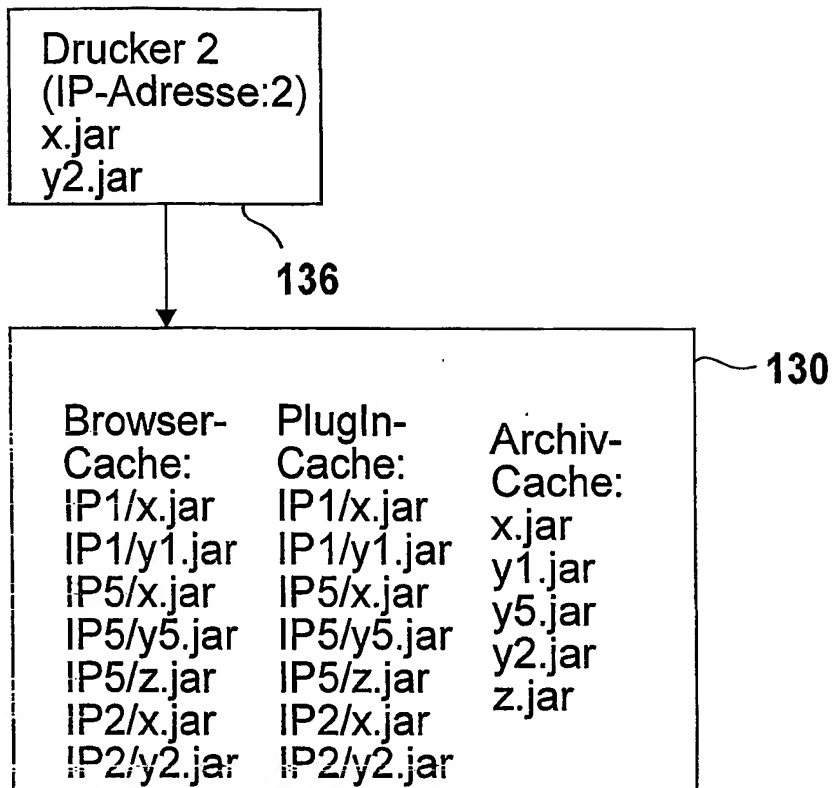


Fig. 9

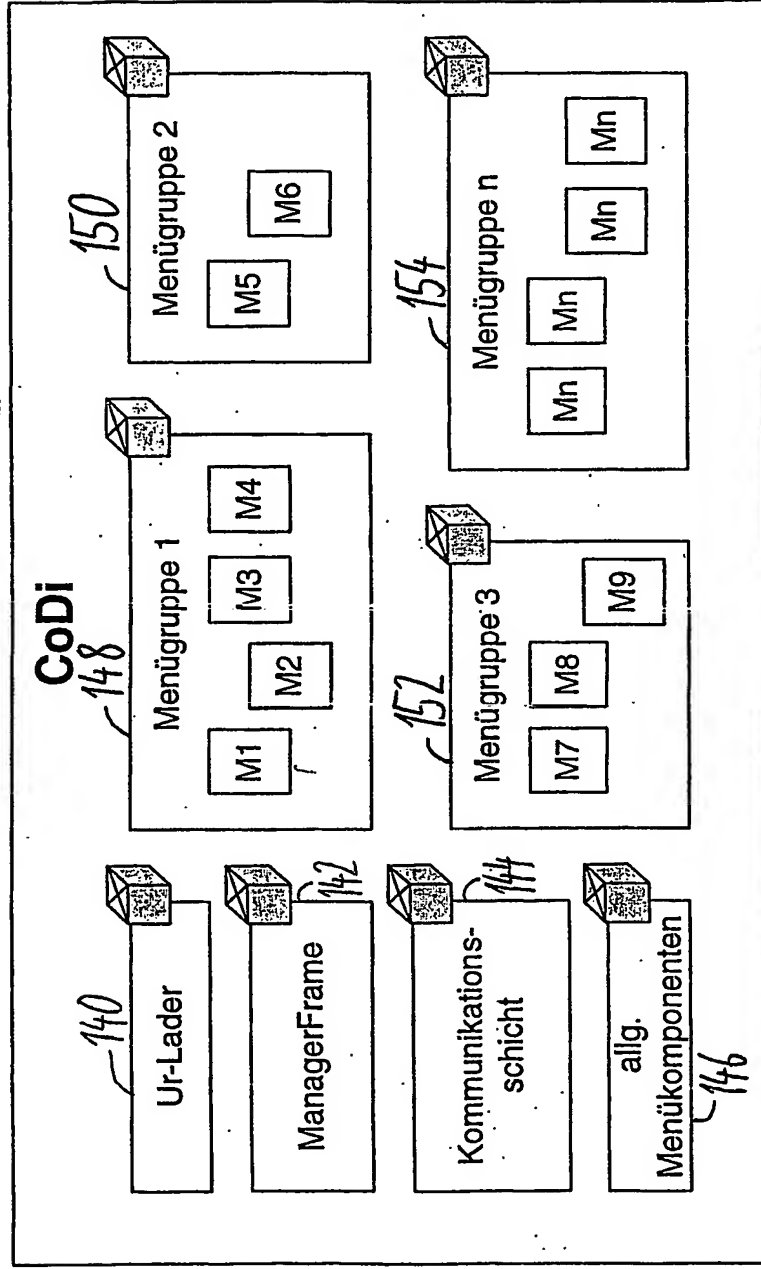


Fig. 10